

**VŠB - Technická univerzita Ostrava**  
**Univerzitní studijní programy**

**Návrh a realizace demonstračního vozítka s řídicím systémem**  
**Arduino**  
**Design and Practical realization of Demonstration Car with**  
**Arduino Control System**

# Zadání bakalářské práce

Student:

**Patrik Červený**

Studijní program:

B3943 Mechatronika

Studijní obor:

3906R007 Automobilová elektronika

Téma:

Návrh a realizace demonstračního vozítka s řídicím systémem Arduino  
Design and Practical realization of Demonstration Car with Arduino  
Control System

Jazyk vypracování:

čeština

Zásady pro vypracování:

- 1) Dle instrukcí vedoucího proveďte rešeršní činnost pro shromáždění širšího rozsahu komponentů, kompatibilních s platformou Arduino, pro pozdější využití při návrhu modelu vozítka pro 3. ročník soutěže Students for Automotive (S4A).
- 2) Navrhněte a prakticky realizujte model soutěžního vozidla s řídicím systémem Arduino, senzory pro možnost autonomní jízdy a rozpoznávání barev a dále s IoT rozhraním pro vzdálené ukládání dat. Konkrétní požadavky na funkčnost vozidla budou specifikovány vedoucím práce.
- 3) Prakticky ověřte funkčnost navržené a realizovaného modelu.
- 4) Navrhněte zadání, technické specifikace a pravidla pro 3.ročník soutěže S4A.

Seznam doporučené odborné literatury:


Dle pokynů vedoucího závěrečné práce.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.


Vedoucí bakalářské práce: **Ing. Petr Šimoník, Ph.D.**

Datum zadání: 01.09.2017

Datum odevzdání: 21.05.2018

  
doc. Ing. Petr Palacký, Ph.D.  
vedoucí katedry



  
Ing. Zdeňka Chmelíková, Ph.D.  
prorektorka pro studium

### **Místopřísežné prohlášení**

Prohlašuji, že jsem celou bakalářskou práci včetně příloh vypracoval samostatně pod vedením Ing. Petra Šimoníka, Ph.D a uvedl jsem všechny použité podklady a literaturu.

V Ostravě dne 21.5.2018

---

Vlastnoruční podpis autora

Prohlašuji, že

- jsem byl seznámen s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména §35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a §60 – školní dílo,
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně ke své vnitřní potřebě bakalářskou práci užít (§35 odst. 3),
- souhlasím s tím, že bakalářská práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího bakalářské práce. Souhlasím s tím, že údaje o bakalářské práci, budou zveřejněny v informačním systému VŠB-TUO,
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu §12 odst. 4 autorského zákona,
- bylo sjednáno, že užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše),
- beru na vědomí, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě dne 21.5.2018

---

Vlastnoruční podpis autora

Jméno a příjmení autora práce

Patrik Červený

Adresa trvalého pobytu autora práce

Royova 782/11

020 01 Púchov

Slovensko

## **Pod'akovanie**

Na tomto mieste by som chcel pod'akovať hlavne pánovi Ing. Petrovi Šimoníkovi, Ph.D za príkladné vedenie a odborné konzultácie bakalárskej práce. Taktiež by som veľmi rád pod'akoval mojej rodine a priateľom za podporu počas celého štúdia.

## **Abstrakt**

Cieľom bakalárskej práce je návrh a realizácia demonštračného autíčka s riadiacim systémom Arduino pre tretí ročník súťaže Students for Automotive. Prvá časť bakalárskej práce sa zaoberá teoretickým rozborom z pohľadu predchádzajúceho druhého ročníka súťaže a následným prieskumom trhu pre možnosti použitia komponentov pre dosku Arduino MEGA 2560 a ďalej rieši teoretický návrh samotného zadania súťaže. Druhá časť je zameraná na praktické zostavenie autíčka podľa vytvoreného zadania a tým demonštruje realizáciu tohto zadania. Pri tom všetkom by mali byť zachované hlavné myšlienky a tými sú autonómnosť a IoT (internet vecí). Výstupom práce je samotné demonštračné autíčko, zadanie súťaže a manuál, ktorý má tímom pomôcť nadviazať komunikáciu medzi jednotlivými časťami autíčka.

## **Kľúčové slová**

Arduino, autíčko, autonómnosť, IoT, diaľkové ovládanie, WiFi, súťaž S4A

## **Abstract**

The aim of the Bachelor Thesis is to design and implement a demonstrational car with the Arduino control system for the third year of the Students for Automotive competition. The first part of the bachelor thesis deals with the theoretical analysis from the perspective of the previous year of the competition, followed by market research of possible uses of the components for the Arduino MEGA 2560 board and by further looks into the theoretical design of the competition itself. The second part focuses on the practical construction of the car according to the given assignment, thus demonstrating its realization. The gist that should be preserved is the autonomy and IoT (the Internet of Things). The output of the work is the demonstration of the car itself, the rules of the competition and a manual for helping the teams to establish communication between the different parts of the car.

## **Key words**

Arduino, Car, Autonomy, IoT, Remote Control, WiFi, S4A Competition

# Obsah

<b>ZOZNAM POUŽITÝCH SKRATIEK A SYMBOLOV .....</b>	<b>9</b>
<b>1 ÚVOD .....</b>	<b>10</b>
1.1 SÚŤAŽ „STUDENTS FOR AUTOMOTIVE“ (S4A).....	10
1.2 SPÄTNÁ VÄZBA.....	10
<b>2 NÁVRH KONCEPCIE SÚŤAŽE S4A III. ....</b>	<b>14</b>
2.1 OBECNÁ KONCEPCIA SÚŤAŽE .....	14
2.2 NÁVRH SÚŤAŽNÝCH DRÁH A ZADANIA .....	14
2.2.1 Sledovanie čiary.....	15
2.2.2 Bludisko .....	15
<b>3 TEORETICKÝ NÁVRH SÚŤAŽNÉHO AUTÍČKA A ĎALŠÍCH PRVKOV .....</b>	<b>17</b>
3.1 NÁVRH KONŠTRUKCIE A OBVODOVÉHO RIEŠENIA SÚŤAŽNÉHO AUTÍČKA .....	18
3.2 ROZBOR POUŽITÝCH KOMPONENTOV .....	19
3.2.1 Podvozok autíčka.....	19
3.2.2 Arduino Mega 2560 Rev3 .....	20
3.2.3 Adafruit Motor Driver Shield L293D .....	22
3.2.4 NodeMCU ESP8266 .....	23
3.2.5 IR senzor YL-70 pre sledovanie čiary.....	24
3.2.6 Zdroj energie.....	25
3.3 ČASOMIERA.....	26
<b>4 PRAKTICKÁ REALIZÁCIA.....</b>	<b>28</b>
4.1 ZOSTAVENIE AUTÍČKA.....	28
4.2 VÝVOJ APLIKAČNÉHO SOFTVÉRU.....	31
4.2.1 Návrh algoritmu pre sledovanie čiary .....	32
4.2.2 Návrh algoritmu pre bludisko.....	35
4.3 VÝVOJ MOBILNEJ APLIKÁCIE.....	36
4.4 FINALIZÁCIA DEMONŠTRAČNÉHO AUTÍČKA .....	40
4.5 ZHOTOVENIE ČASOMIERY .....	41
<b>5 VYHLIADKY DO BUDÚCNOSTI .....</b>	<b>43</b>
<b>ZÁVER .....</b>	<b>44</b>
<b>ZOZNAM POUŽITEJ LITERATÚRY .....</b>	<b>45</b>
<b>ZOZNAM POUŽITÝCH OBRÁZKOV .....</b>	<b>46</b>

<b>ZOZNAM VÝPISOV ZDROJOVÉHO KÓDU.....</b>	<b>47</b>
<b>PRÍLOHY .....</b>	<b>48</b>
A    FOTOGRAFIE AUTÍČKA.....	48
B    FOTOGRAFIE ČASOMIERY.....	49
C    OBSAH CD .....	50



## Zoznam použitých skratiek a symbolov

AP	Access Point (Prístupový bod)
App	Application Software (Aplikačný softvér)
DC	Direct Current (Jednosmerný prúd)
EEPROM	Electrically Erasable Programmable Read-Only Memory (Elektricky vymazateľná pamäť len pre čítanie)
FEI	Fakulta elektotechniky a informatiky
GND	Ground (Zem elektrického obvodu)
HTTP	Hypertext transfer protocol (Hypertextový prenosový protokol)
IDE	Integrated Development Environment (Integrované vývojové prostredie)
ISP	In System Programmer (Možnosť programovať mikrokontrolér priamo na doske)
IoT	Internet of Things (Internet vecí)
I/O	Input/Output (Vstupné/Výstupné napríklad periférie)
IP	Internet Protocol (Internetový Protokol)
IR	Infra Red (Infračervené)
LASER	Light Amplification by Stimulated Emission of Radiation (Zosilňovanie svetla stimulovanou emisiou žiarenia)
LED	Light emitting diode (Elektro-luminiscenčná dioda)
MT	Mobilný Telefón
OS	Operating System (Operačný systém)
PWM	Pulse Width Modulation (Pulzne šírková modulácia)
USB	Universal Serial Bus (Univerzálna sériová zbernica)
URL	Uniform Resource Locator (Jednotný vyhľadávač prostriedku)
ICSP	In-Circuit Serial Programming (Obvodové sériálové programovanie)
FLASH	(Pamäť pre ukladanie a prenos dát medzi počítačom a digitálnym zariadením)
SRAM	Static Random Access Memory (Statická prepisovateľná pamäť)
UART	Universal Asynchronous Receiver/Transmitter (Fyzický elektrický obvod pre univerzálnu asynchrónnu sériovú komunikáciu)
SoC	System on Chip (systém integrujúci komponenty počítača do jedného čipu)
SPI	Serial Peripheral Interface (Sériové periférne rozhranie)
TWI	Two Wire Interface (Dvojkáblové rozhranie)

# 1 Úvod

## 1.1 Súťaž „Students for Automotive“ (S4A)

Vysoká škola báňská - Technická univerzita Ostrava, Fakulta elektrotechniky a informatiky (ďalej len FEI), Katedra elektroniky (KAT430), sekcia Automotive opakovane vyhlasuje súťaž o najlepšie autíčko postavené na platforme Arduino.

Súťaž S4A je výsledkom riešenia projektu s názvom „Motivační soutěže v technických oborech“, ktorý vytvára priestor pre popularizáciu techniky a zapojenie študentov do súťaží, ktoré komplexným spôsobom kombinujú niekoľko oborov. Hlavnou myšlienkou je zvýšiť záujem študentov o tzv. „bastlenie“ v oblasti automotive. V dnešnej dobe hotových výrobkov je podstata súťaže ísť ku koreňom a vytvoriť vlastný produkt od začiatku. Súťaž je cielená predovšetkým na študentov stredných škôl druhých a tretích ročníkov maturitných odborov. Tímy sú zložené z maximálne troch študentov a jedného dohliadajúceho pedagóga. Z pôvodne zamýšľaných 8 tímov, sa druhého ročníka súťaže nakoniec zúčastnilo tímov 17.

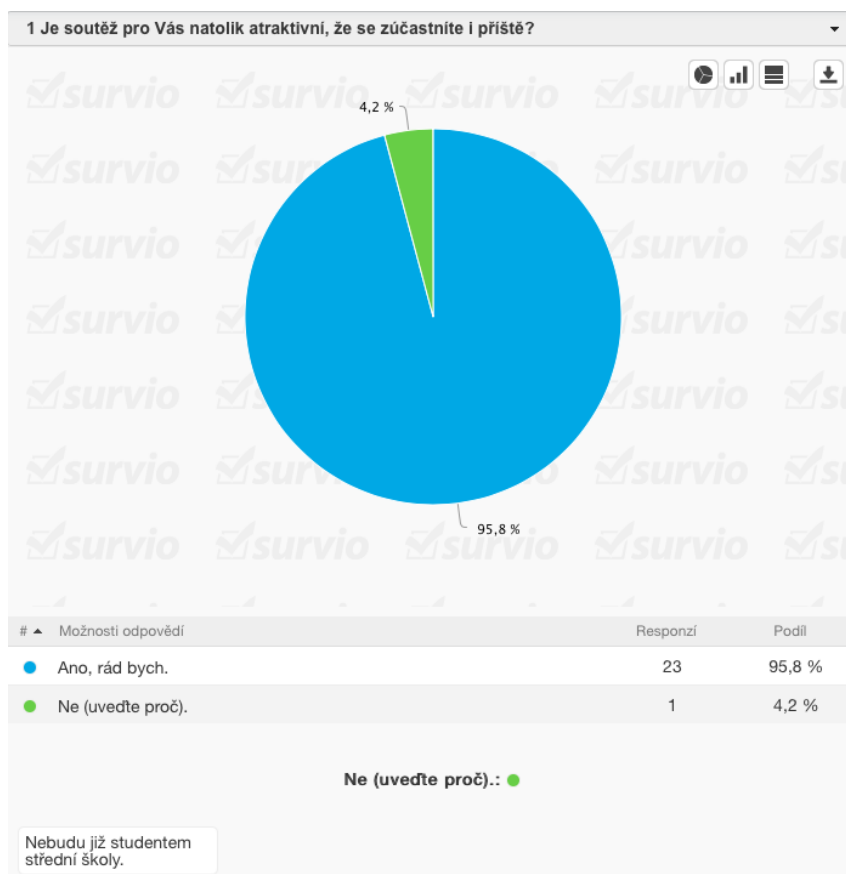
Základom je postaviť autíčko na robotickom podvozku s náhonom na všetky kolesá. Riadenie a pohyb autíčka budú zaistené vývojovou doskou Arduino Mega 2560 a príslušnými komponentmi. Túto vývojovú dosku bude nutné naprogramovať, tak aby všetky komponenty spolupracovali ako jeden celok. Takto vytvorené autíčko absolvuje súťažné jazdy na vopred stanovených súťažných dráhach. Súčasťou súťaže je prezentácia realizácie autíčka pred ostatnými tímami.

Hodnotené sú kategórie: kvalita prevedenia zostavenia autíčka, doba prejazdu súťažnými dráhami a osobná prezentácia realizácie autíčka. Hodnotenie každoročne prebehne pri slávnostnom finále na FEI v Ostrave, v Apríli 2019, za účasti zástupcov generálnych partnerov súťaže spoločnosti Allgeier Engineering Czech s.r.o, HELLA AUTOTECHNIK NOVA s.r.o a portálu HWKITCHEN.CZ. Za každú kategóriu budú porotou udelené body. Pre zúčastnené tímy sú pripravené hodnotné ceny, ktoré do súťaže každoročne venujú vyššie spomenutí partneri.

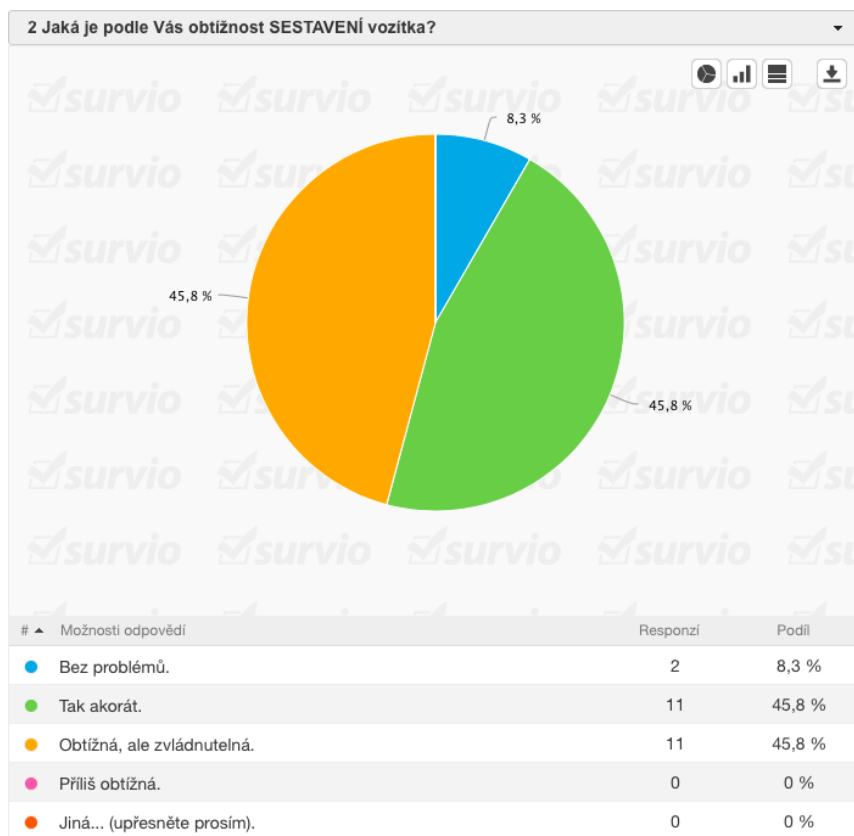
## 1.2 Spätná väzba

Pre zlepšenie tretieho ročníka súťaže Students for Automotive bol vytvorený online dotazník, ktorý mal slúžiť, ako spätná väzba od účastníkov druhého ročníka súťaže. Dotazník bol vytvorený na webovej stránke [www.survio.com](http://www.survio.com), bol anonymný a obsahoval 5 otázok, ktorých vyplnenie zabralo niekoľko minút. Z celkových 51 účastníkov súťaže vyplnilo dotazník 24. Webová stránka [www.survio.com](http://www.survio.com) podporuje analýzu výsledkov

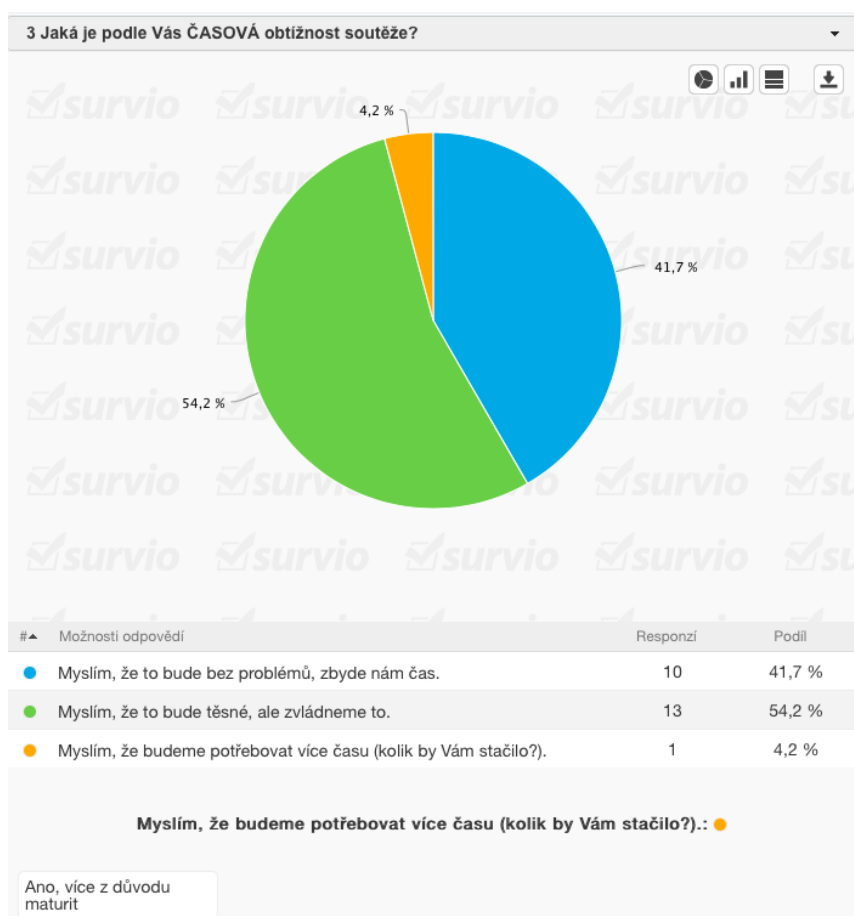
v podobe grafov, vid' nasledujúce obrázky. Vďaka podnetom v poslednej otázke zobrazenej na obrázku 1.5, bolo pri tvorbe zadania nadväzujúceho ročníka súťaže rozhodnuté použiť WiFi modul NodeMCU ESP 8266, ktorý slúži ako bezdrôtový prenos dát medzi doskou Arduino Mega 2560 umiestnenou na autíčku a MT s OS Android, ktorým sa dá autíčko ovládať. Tieto komponenty sú opísané v kapitole 3.2.



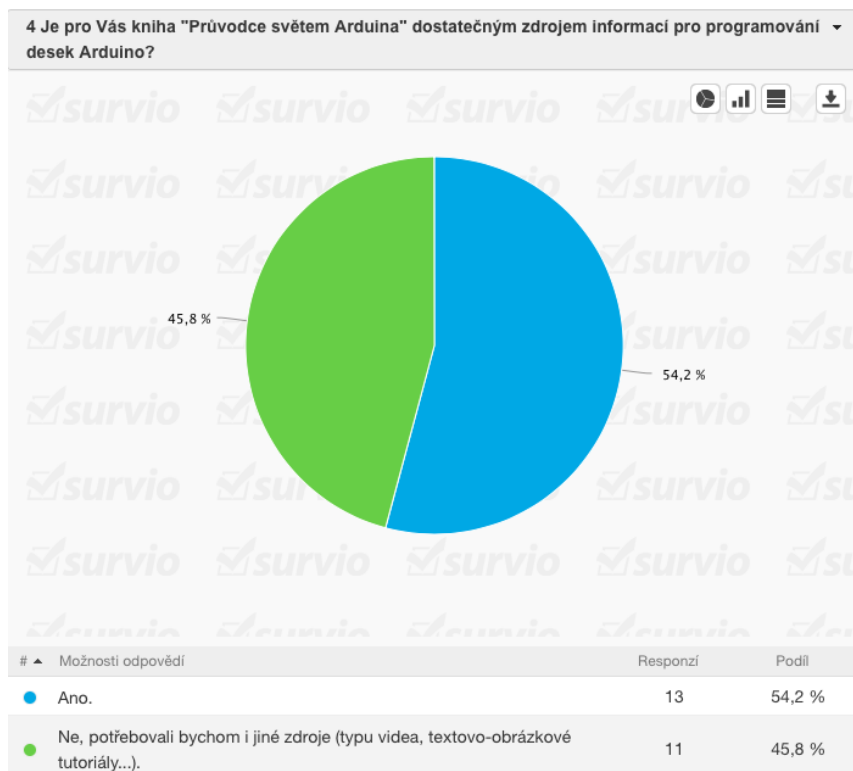
Obrázok 1.1. Online Dotazník - otázka č.1



Obrázok 1.2. Online Dotazník - otázka č.2



Obrázok 1.3. Online Dotazník - otázka č.3



Obrázok 1.4. Online Dotazník - otázka č.4

5 Jaké jsou Vaše tipy a kritické připomínky pro zlepšení dalšího ročníku soutěže?

Lepší výstupní kontrola (držák na baterie neměl 2 kontakty s pružinami, museli jsme opravit), jednoznačnější zadání - na shledu a některých komponentách je nutné napájet PINY pro připojení kablíků, nebyli jsme si 100% jistí, zda pájení na desce je povolený úkon.

Myslím, že soutěž je připravená opravdu výborně. Málo kde Vám nabídnou tolik zdrojů informací a pomoci, jako tady, včetně samotného materiálu. I ceny jsou opravdu lákavé. Ovšem počet soutěžících týmů by mohl být menší. Vzhledem k tomu, že mají všichni stejné zadání i prostředky, tak to činí soutěž velmi náročnou na hodnocení ze strany poroty i zpracování ze strany soutěžících. Jinak jsem se soutěží, jako takovou, spokojený.

Nějaká aplikace s ESP8266.

zatím nic

Více vanity dílu. Proste třeba nějaký reprak a tak at si s roborem muzeme vyhrat

Některé součástky přišli poškozené a bylo potřeba je přepájet. Nelíbí se mi, že do hodnocení bude zahrnut i vizuální vzhled autíčka.

K baterkám není nabíječka.

Mohla by být mnohem lepší organizace soutěže a přesnější zadání úkolů (sledování dráhy, výška bludiště atd.)

Žádné

Žádné!

Letos bez připomínek.

- (2x)

zatím žádné připomínky nemám

Pomohlo by přidat dokumentaci k jednotlivým komponentům, popř. i krátké příklady, jak se tyto senzory a komponenty ovládají.

Zatím zadne.

N

Jsem rád, že soutěž je. Kritické připomínky nemám.

Můj tip je, aby soutěžící měli lepší představu o bludišti (např. aby plánec bludiště byl nakreslen i ve 3D prostoru)

Více komponent k sestavení auta.

Bylo by vhodné, kdyby měl každý tým přidělen konzultanta, který by poskytoval v případě problémů rady

Žádné nemám

nevím

Obrázok 1.5. Online Dotazník - otázka č.5

## **2 Návrh koncepcie súťaže S4A III.**

Princípom súťaže je postaviť autíčko s pohonom všetkých štyroch kolies, ktoré je schopné prejsť jasne definovanými súťažnými dráhami. Tímom budú dodané vopred zvolené komponenty, pričom bez požiadania organizátora ich množstvo nemôže byť rozšírené. Voľba týchto komponentov bola prevedená na základe zadania súťaže, ktorého definícia je opísaná vo zvyšku tejto kapitoly. Presné znenie tohto zadania, spoločne s podmienkami súťaže pre ďalší ročník S4A budú definované a uverejnené pred vyhlásením súťaže.

### **2.1 Obecná koncepcia súťaže**

Súťaž je navrhnutá tak, že zúčastnené tímy dostanú k dispozícii zadanie súťaže, komponenty a návod, ktorý im má pomôcť zostrojiť autíčko tak, aby správne prebiehala komunikácia medzi jednotlivými časťami autíčka. Ďalej je na nich naprogramovať autíčko takým spôsobom, aby čo najlepšie splnili zadanie súťaže. Na celú prípravu majú vyhradený vopred stanovený časový úsek. Počas prípravy bude medzi organizátorom a každým tímom prebiehať komunikácia ako podpora pre tímy a zároveň spätná väzba pre organizátora. Po uplynutí tohoto času sa v Apríli 2019 tímy stretnú v priestoroch FEI, kde prebehne závod S4A. Presný dátum bude včas oznámený.

Závodný deň pozostáva z nasledujúcich častí:

- zahájenie súťaže v podobe prezentácie súťaže S4A a FEI zo strany organizátora,
- tímy prezentujú počítačovú prezentáciu, ktorá zahŕňa dokumentáciu o tvorbe a vývoji ich autíčka,
- tímy súťažia na súťažných dráhach,
- vyhodnotenie a ukončenie súťaže.

Hodnotenú kritéria sú nasledovné:

- doba prejazdu jednotlivými dráhami,
- počítačová prezentácia a vystupovanie,
- kvalita a precíznosť prevedenia praktickej realizácie autíčka,
- dizajn autíčka.

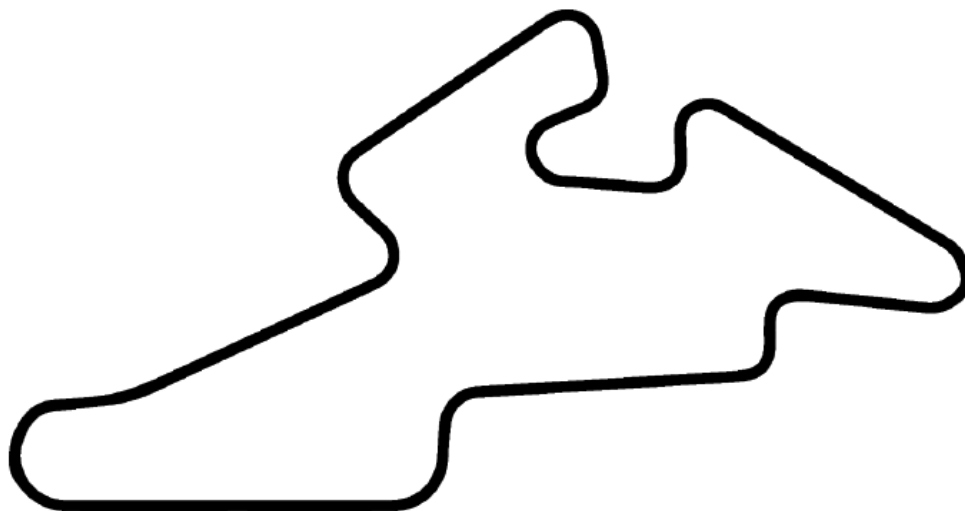
### **2.2 Návrh súťažných dráh a zadania**

Hlavnými témami je autonómnosť a IoT. Na základe toho musela byť definícia zadania a voľba použitých komponentov vykonaná tak, aby boli zachované obe tieto témy. Zároveň súťažné dráhy by nemali byť komplikované tak, aby autíčko, ktoré má nimi prejsť boli

schopní vytvoriť aj žiaci stredných škôl pre ktorých je súťaž určená. Navrhnuté súťažné dráhy sú opísané v nasledujúcich kapitolách.

### 2.2.1 Sledovanie čiary

Ako je možné vidieť na obrázku 2.1, tvar súťažnej dráhy je v podobe Masarykovho okruhu v Brne v mierke 1:600.

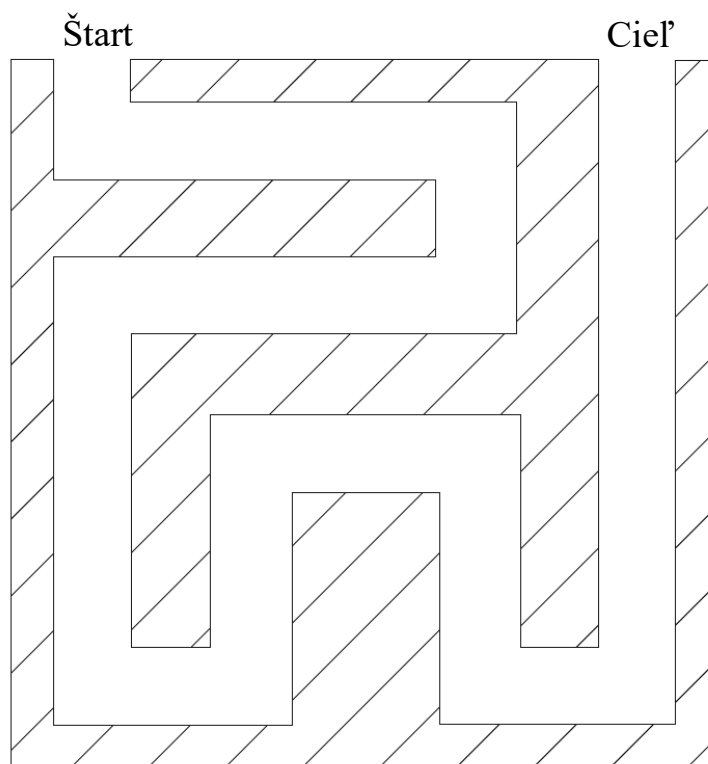


*Obrázok 2.1. Súťažná dráha - sledovanie čiary (Masarykov okruh)*

Táto dráha bola vytvorená pomocou čiernej lepiacej pásky, ktorá je určená priamo pre tento účel - páska veľmi dobre pohlcuje IR žiarenie. Dráha bola umiestnená na kontrastný biely povrch tvorený doskami sololitu. Tento povrch naopak, veľmi dobre odráža IR žiarenie. Na základe tohto javu bol obsahom dodávaných komponentov aj IR senzor prekážok *YL-70*, určený práve pre praktickú realizáciu robotov, ktorý nasledujú čiaru na kontrastnom povrchu. Dráha prezentuje myšlienku autonómnej jazdy, bez zásahu užívateľa. Autíčko pomocou spomínaného senzora sníma povrch po ktorom jazdí a na základe toho ktorý snímač IR žiarenia sníma čiaru, vykoná príslušnú akciu, napríklad zatočí do strany. Senzor prekážok je opísaný v kapitole 3.2.5.

### 2.2.2 Bludisko

Pôdorys súťažnej dráhy je zobrazený na obrázku 2.2.



*Obrázok 2.2. Súťažná dráha - bludisko*

Mantinely dráhy sú vysoké 20 cm a vnútorná šírka dráhy je 36 cm. Ďalej má jasne definovaný štart a cieľ pričom neobsahuje žiadne slepé cesty. Podložka na ktorej je dráha umiestnená pozostáva rovnako ako v prípade predchádzajúcej dráhy z bielych sololitových dosiek. Mantinely sú tvorené drevenými doskami.

Súťažné tímy majú za úlohu prejsť túto dráhu diaľkovo ovládaným autíčkom. Ako diaľkový ovládač budú musieť použiť MT s OS Android. Komunikácia medzi autíčkom a MT bude prebiehať pomocou bezdrôtovej siete WiFi. Tento OS bol zvolený práve pre fakt, že sa jedná o otvorenú platformu a existuje veľa možností, ako možno vytvárať mobilné aplikácie pre tento systém. Praktická realizácia tohoto návrhu je opísaná v kapitole 4.2.2.



### 3 Teoretický návrh súťažného autíčka a ďalších prvkov

Základom celej koncepcie je platforma Arduino, ktorá bola zvolená pre súťaž práve pre jednoduché vytváranie komplexných projektov. Existuje nespočetné množstvo I/O periférií v podobe snímačov fyzikálnych či chemických veličín, klávesníc, displejov, motorov, modulov pre rozšírenie, takzvaných shieldov (ďalej len shield) a ďalších komponentov, ktoré sú na trhu ľahko dostupné a pripravené priamo pre pripojenie k doske Arduino.

#### O Arduine

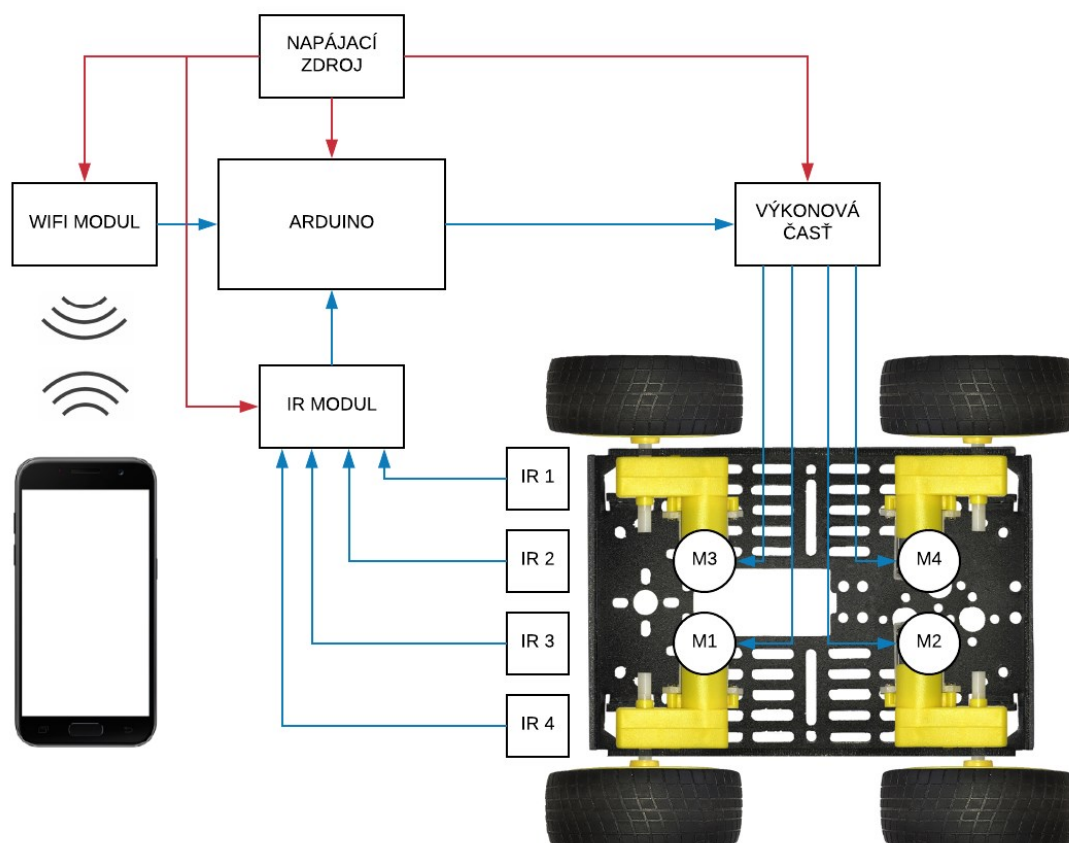
Arduino je otvorená elektronická platforma, založená na užívateľsky jednoducho použiteľnom hardware a software. Dosky Arduino sú schopné čítať vstupné periférie, ako napríklad svetlo dopadajúce na senzor, prst na tlačidlo alebo správu na Twitter a premeniť túto informáciu na výstup - aktivovanie motora, rozsvietenie LED alebo publikovať niečo online. Pomocou zaslania série inštrukcií do mikrokontroléra umiestneného priamo na doske, je možné doske prikázať, akú akciu má vykonať. To sa dá urobiť pomocou programovacieho jazyka Arduino IDE založeného na programovacom jazyku pre mikrokotroléry Wiring, ktorý je založený na *Processing*. V priebehu rokov, bolo Arduino použité ako “mozog” tisícov projektov od každodenných objektov až po komplexné vedecké nástroje. Na tejto otvorenej platforme sa zhromaždila svetová komunita tvorcov - študenti, kutili, umelci, programátori a profesionáli, ktorých príspevky prispeli k neuveriteľnému množstvu poznatkov, ktoré môžu byť pre začiatočníkov i profesionálov veľmi nápomocné. [1]

Vývoj prvého Arduina začal v roku 2005, keď sa ľudia z talianskeho Interaction Designe Institute v meste Ivrea rozhodli vytvoriť lacný vývojový set pre študentov, ktorí si nechceli zaobstarávať, v tej dobe rozšírené a drahé dosky BASIC Stamp. Medzi študentmi sa Arduino uchytilo a tak sa tvorcovia rozhodli poskytnúť ho celému svetu. A to nie len predajom vlastných dosiek, ale aj zdieľaním všetkých schém a návodov (jedná sa o *open-source* projekt, ďalej len *open-source*).

Programová časť Arduina bola založená na *Processing*, čo je knižnica pre jazyk Java s vlastným editorom. Všetko ma za cieľ zjednodušiť výuku programovania. V dnešnej dobe sa predalo viac ako už niekoľko stotisíc dosiek Arduino. Dôkazom, že táto platforma nie je mŕtva, môže byť aj to, že nedávno bol ohlásený vývoj novej a výkonnej dosky Arduino Galileo, ktorý vzniká v spolupráci so spoločnosťou Intel. Za osem rokov vývoja už vzniklo mnoho rôznych typov Arduina. Keďže sa jedná o *open-source* projekt, vznikalo spoločne s hlavnou líniou projektu aj mnoho ďalších neoficiálnych typov, takzvaných klonov. [2]

### 3.1 Návrh konštrukcie a obvodového riešenia súťažného autíčka

Konštrukcia autíčka vychádza zo zadania súťaže, ktoré bolo definované v kapitole 2. Na základe toho bolo nutné vytvoriť návrh riešenia, ktoré by odpovedalo požiadavkám na výslednú funkcionality demonstračného autíčka. Takto navrhnutý celok je zobrazený na obrázku 3.1.



Obrázok 3.1. Blokova schéma navrhnutého demonstračného autíčka

Doska mikrokontroléra je riadiaca časť elektronických systémov autíčka, a zohráva dominantnú úlohu v celej koncepcii. Prijíma informácie od pripojených vstupných periférií, logicky ich spracováva podľa definovaného programu a následne tieto informácie zasiela v podobe inštrukcií do výkonovej časti. Na výstup tejto výkonovej časti sú pripojené ovládané akčné členy (DC motory). Definovanie týchto pohybov prebieha na základe spracovania informácií, ktoré sú dostupné na vstupných perifériách pripojených k Arduinu.

Pre súťažnú dráhu *Sledovanie čiar*, pohyby autíčka určujú snímače IR žiarenia, ktoré musia byť schopné toto žiarenie vyslať a následne ho po odraze od povrchu spätne snímať. Pri dopade IR žiarenia na čiernu pásku sa toto žiarenie z väčšej časti prípadne úplne pohltí a teda neodráža späť do prijímacej časti snímača. Tak mikrokontrolér od tohto snímača dostane informáciu o tom, že daný snímač sa nachádza nad páskou. Podľa toho o ktorý

snímač sa jedná, mikrokontrolér zašle inštrukcie pre požadovanú akciu do výkonovej časti. Návrh algoritmu pre spracovanie informácií získavaných zo snímačov IR žiarenia je detailnejšie opísaný v kapitole 4.2.1.

Pre súťažnú dráhu *Bludisko* pohybu autíčka definuje užívateľ prostredníctvom diaľkového ovládania. Na MT s OS Android pomocou vytvorenej mobilnej aplikácie zadáva požiadavky na pohyb autíčka. Tieto požiadavky sú zasielané prostredníctvom bezdrôtovej WiFi siete do WiFi modulu, odkiaľ sú následne zasielané do Arduina. Ako už bolo spomenuté, Arduino zašle príkazy do výkonovej časti a tá ovláda jednotlivé motory podľa týchto príkazov. Realizácia tejto komunikácie je opísaná v kapitole 4.2.2.

Na základe tejto koncepcie sú v nasledujúcich kapitolách opísané jednotlivé zvolené komponenty, ktoré sú schopné splniť požadované nároky.

### **3.2 Rozbor použitých komponentov**

Ako je opísané v úvode kapitoly 3 existuje mnoho komponentov, ktoré môžu spolupracovať s doskami Arduino. Preto je na uvážení každého, kto by mohol vytvárať model autíčka, ktoré z ponúkaných komponentov si zvolí tak, aby autíčko bolo schopné splniť zadanie súťaže. Z toho vyplýva, že nie je zrejmé ktoré riešenie je správne. Pre tento fakt boli pri voľbe komponentov zahrnuté nasledujúce požiadavky:

- každý komponent musí byť jednoducho ovládateľný,
- výsledný celok musí byť čo najviac prehľadný a jednoducho zostaviteľný,
- nadviazať komunikáciu medzi jednotlivými komponentmi nesmi byť komplikované,
- ideálne je vybrať komponenty, pre ktoré sú dostupné knižnice.

Práve spomínané knižnice uľahčujú prácu, skracujú čas potrebný na naprogramovanie jednotlivých komponentov a dodávajú na prehľadnosti finálneho kódu. Nasledujúce podkapitoly opisujú jednotlivé komponenty, ktoré boli v rámci riešenia tejto bakalárskej práce zvolené pre splnenie vytvoreného zadania súťaže.

#### **3.2.1 Podvozok autíčka**

Ako základňa autíčka bola použitá platforma SparkFun Multi-chassis 4WD Kit (Basic). Tento set sa dá veľmi jednoducho skladať a používať, poskytuje rýchlu platformu s pohonom všetkých štyroch kolies a dostatkom priestoru pre množstvo senzorov a ovládačov. Stačí pridať ovládač motorov, zdroj energie a autíčko je pripravené prevádzky. Set taktiež obsahuje 4 DC prevodovky s prevodom 48:1, ktoré nezávisle roztáčajú všetky 4 nízkoprofilové kolesá s priemerom 65mm. Každá strana hliníkového rámu s hrúbkou 2,5

mm je vybavená množstvom upevňovacích bodov, ktoré umožňujú uchytenie ovládačov a senzorov autíčka.

Špecifikácie setu:

- 4x DC motor s krútiacim momentom 78.45 Nm,
- odporúčané napätie 4,5 V,
- nominálny prúd bez záťaže 190 mA (maximálne 250 mA),
- otáčky bez záťaže 140 ot./min,
- rozmery setu: dĺžka 165 mm, šírka 157 mm, výška 65 mm. [3]



Obrázok 3.2. Použitý podvozok 4WD

[3]

### 3.2.2 Arduino Mega 2560 Rev3

Arduino Mega 2560 Rev3 je *open-source* platforma založená na mikrokontroléri ATmega2560. Je konštruovaná pre komplexnejšie projekty, preto je na doske dostupných 54 digitálnych I/O pinov (z čoho 15 môže byť použitých ako výstup s PWM), 16 analógových vstupov, každý poskytuje 10 bitové rozlíšenie (1024 rôznych hodnôt), väčší priestor pre program, preto je doska odporúčaná pre použitie v 3D tlačiarňi a podobných robotických projektoch. Poskytuje kopec priestoru a možností pre tvorbu komplexných projektov. Ďalej obsahuje 4 UART porty, 16 MHz kryštálový oscilátor, USB pripojenie, konektor napájania, ICSP konektor a reset tlačidlo. Pre napájanie dosky je možné použiť

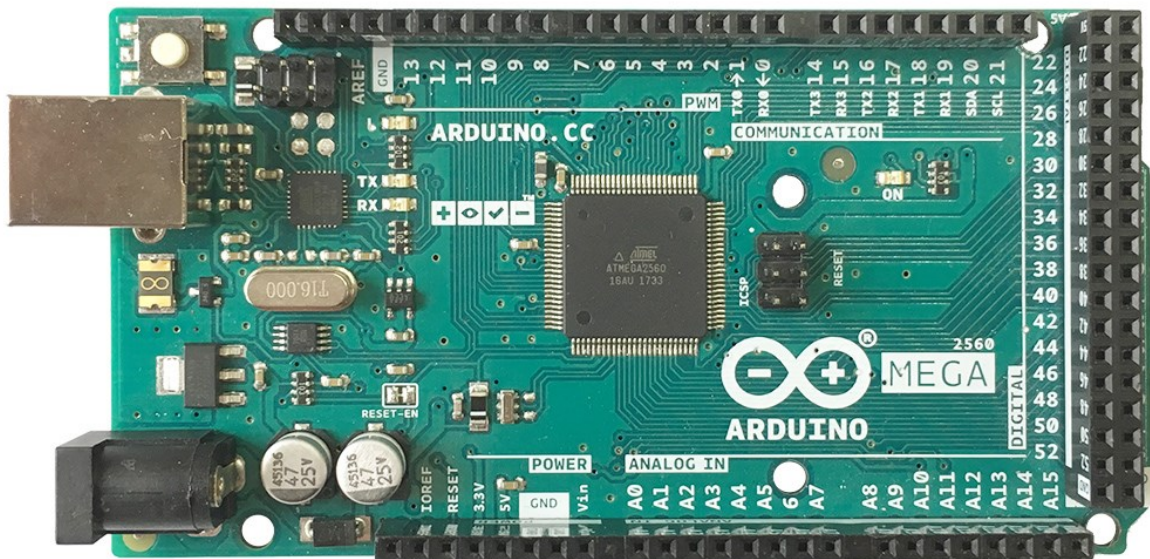
USB konektor alebo externý zdroj (najmä AC/DC adaptér a batéria) pričom napájací zdroj je volený automaticky.

Doska môže byť programovaná pomocou softvéru *Arduino IDE*. Mikrokontrolér ATmega 2560 na doske je predprogramovaný bootloaderom, čo umožňuje zaslať do mikrokontroléra nový kód bez nutnosti použiť externý programovací hardvér. Komunikácia prebieha cez protokol STK500 (typ komunikačného protokolu). Taktiež je možné vynechať bootloader a programovať mikrokontrolér priamo cez ICSP piny pomocou Arduino ISP alebo podobného hardvéru.

Základné vlastnosti dosky:

- operačné napätie 5 V DC,
- napájacie napätie 7 až 12 V (odporúčané), 6 až 20 V (limitné),
- jednosmerný prúd 20 mA na I/O pinoch,
- I/O piny majú zabudovaný pull-up rezistor 20 až 50 k $\Omega$  (predvolene vypnutý),
- pamäť Flash 256 kB pre uloženie kódu z toho 8 kB používa bootloader,
- SRAM 8 kB,
- EEPROM 4 kB, môže byť čítaná/zapísaná pomocou EEPROM knižnice,
- ochrana USB portu proti skratu a nadprúdu (nad 500 mA),
- 4x pin pre sériovú komunikáciu RX0-3 a TX0-3,
- 5x pin pre externé prerušenie,
- piny pre SPI komunikáciu,
- vstavaná LED pripojená na pin 13 (keď log. úroveň = 1 LED svieti),
- piny pre TWI komunikáciu,
- Arduino IDE zahŕňa sériový monitor pre jednoduché posielanie textových dát do a z dosky,
- jednoduché uchytenie dosky o povrch alebo puzdro pomocou pripravených otvorov,
- doska je konštruovaná pre väčšinu shieldov vytvorených priamo pre Arduino Uno.

[4]



Obrázok 3.3. Arduino Mega 2560

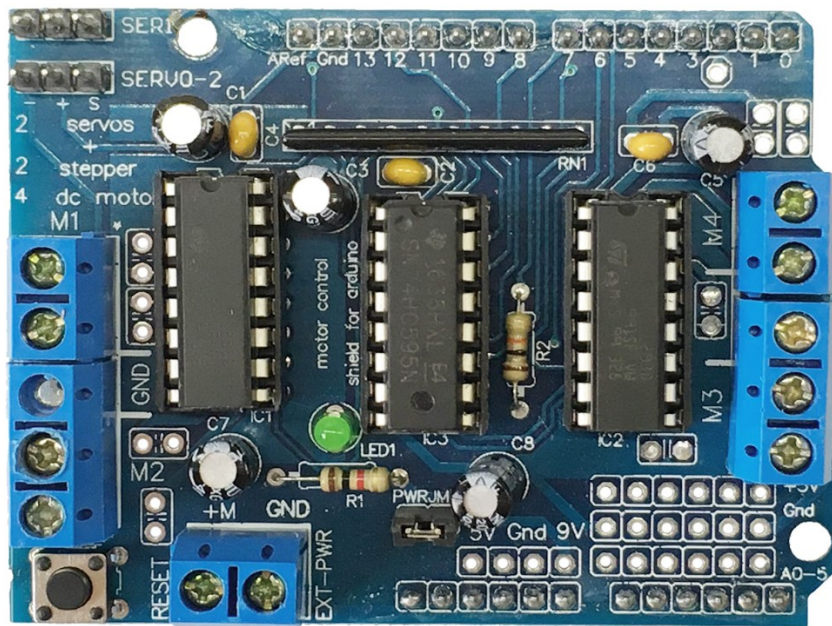
### 3.2.3 Adafruit Motor Driver Shield L293D

*Adafruit Motor driver Shield L293D* je prídavný modul pre Arduino, ktorý je určený pre pohodlné a bezpečné ovládanie troch druhov motorov. Jedná sa o jednosmerné (DC) motory, servo motory a krokové motory, pričom naraz je možné pripojiť dva servo motory a až štyri DC alebo 2 krokové motory. Konektory pre DC a krokové motory sú umiestnené po stranách motor shieldu označené ako M1, M2, M3 a M4. Pre pripojenie krokového motora je nutné využiť dvojicu konektorov vedľa seba, teda M1+M2 alebo M3+M4. Samostatne na dlhšej strane shieldu sa nachádza konektor externého napájania. Vzhľadom k prúdovej náročnosti je nutné oddeliť napájanie shieldu a Arduina práve cez spomínaný konektor externého napájania. Shield je konštrukčne navrhnutý tak, aby sa dal priamo zastrčiť do kompatibilných dosiek Arduino (Uno, Leonardo...) a tým šetrí priestor a čas pri zostavovaní systému. Toto riešenie zaberie väčšinu pinov na doske Arduino. Pre správnu funkciu shieldu je nutné do programovacieho softvéru Arduino IDE importovať knižnicu „Motor\_shield\_library, ktorá zároveň zjednodušuje a urýchľuje prácu so shieldom.

Základné vlastnosti dosky:

- možné pripojiť 2 servo motory a 4 DC alebo 2 krokové motory,
- rozsah napájacieho napätia riadiaceho obvodu L293D je 4,5 až 25 V,
- menovitý prúd na výstupe 0,6 A (1,2 A špičkový),
- tepelná ochrana riadiacich obvodov,
- jednoduché programovanie vďaka knižnici. [5]





Obrázok 3.4. Adafruit Motor Driver Shield

### 3.2.4 NodeMCU ESP8266

NodeMCU je *open-source* platforma, ktorá umožňuje vytvárať interaktívne projekty najmä v oblasti automatizácie či iné projekty zamerané na prepojenie s internetom/inými zariadeniami. ESP8266 je možné nazvať aj SoC. Ponúka všetko, čo bežný vývojár potrebuje od širokých možností programovania cez hardvérové vybavenie po možnosti pretaktovania procesora.

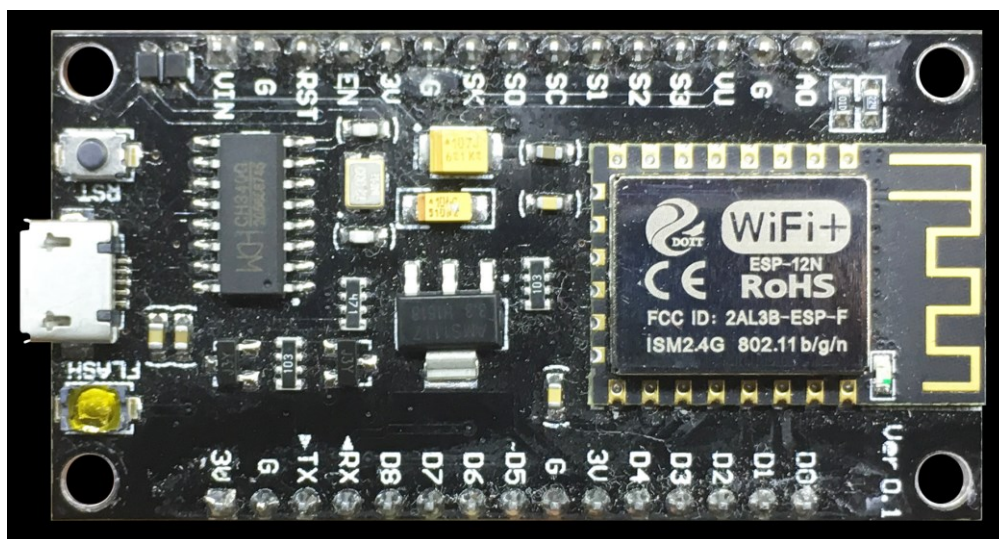
ESP8266 - srdce NodeMCU obsahuje jednojadrový 32 bitový procesor Xtensa LX 106 s inštrukčnou sadou RISC a taktom 80 MHz, pričom je ho možné pretaktovať až na 160 MHz.

Do dosky je možné nahráť až 1 MB skompilovaného kódu, pre porovnanie Arduino Uno iba 32 kB. NodeMCU je vybavené 4 MB Flash pamäťou.

Doska ponúka až 11 GPIO pinov, pričom všetky okrem GPIO16 podporujú PWM. Výstupný prúd GPIO pinov je 12 mA pri 3,3 V. Piny nie sú 5 V tolerantné, i keď niektoré zdroje uvádzajú, že sú. Pri dlhodobom zaťažení na 5 V dôjde k prehriatiu čipu ESP8266 a jeho následnému zničeniu. Na doske sa nachádza iba jeden analógový pin. Štandardne ESP8266 vie pin využívať v rozmedzí 0–1 V, avšak vďaka NodeMCU súčiastkam pracuje v rozmedzí 0–3,3 V. ESP8266 v sebe združuje aj implementáciu WiFi štandardu 802.11 na 2,4 GHz v pásmach b/g/n, pričom v pásme b má garantovaný najlepší zisk 25 dB.

NodeMCU podporuje štandardné rozhrania ako OneWire, I2C, SPI, UART. Prvotná verzia NodeMCU dosiek mala nový firmware a spôsob programovania. Prvé dosky mali NON-OS

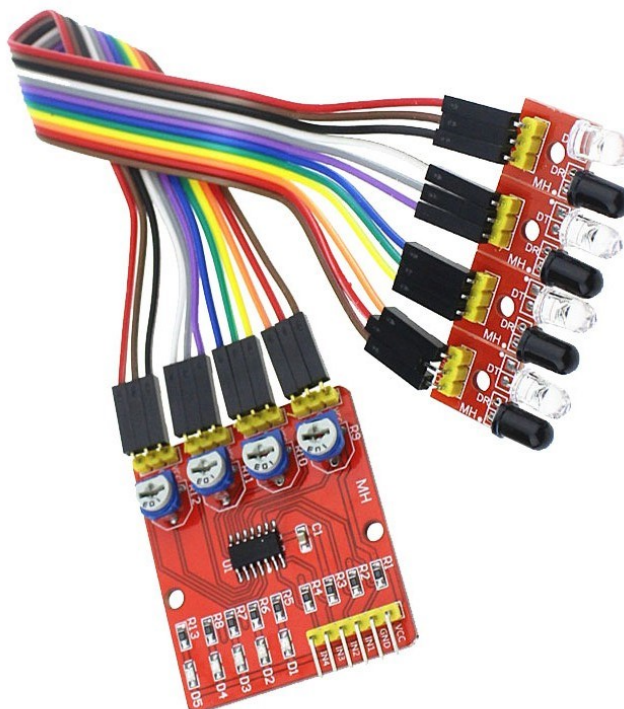
Pre programovanie dosky pomocou vývojového prostredia Arduino IDE je nutné nainštalovať potrebné knižnice, tak ako to opisuje manuál pre žiakov v prílohe na CD.





Technické špecifikácie:

- 4x modul s vysielacou a prijímacou diódou IR žiarenia,
- prevádzkové napätie 3,3 alebo 5 V DC,
- prevádzkový prúd <1A,
- montážne otvory: skrutka M3,
- detekcia prekážky od 1 mm do 60 cm (pri bielej farbe najlepší odraz). [7]



Obrázok 3.6. Senzor prekážok YL-70

[8]

### 3.2.6 Zdroj energie

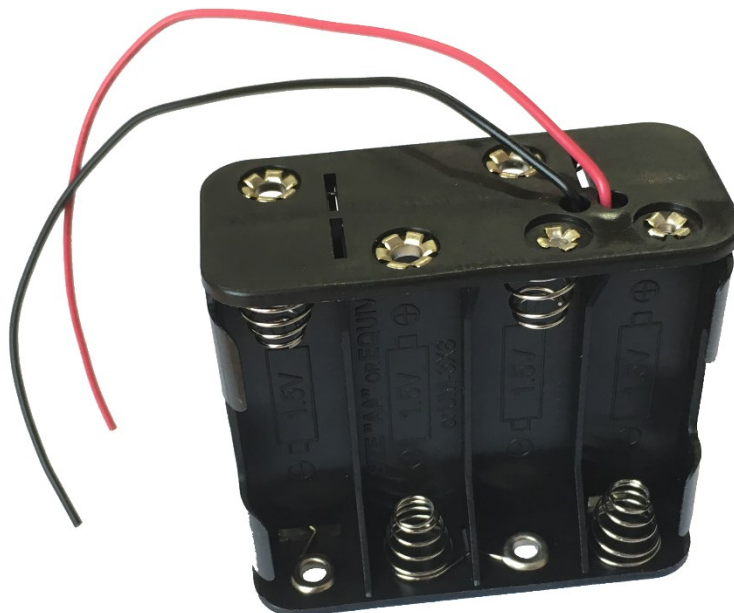
Ako je možné vidieť v kapitole 3.2.2 odporúčané napájacie napätie pre Arduino je 7 až 12 V. Na základe tohto faktu bol po prieskume trhu ako vhodný zdroj energie zvolený batériový zdroj, ktorý sa skladá z ôsmych sériovo zapojených batérií typu AA. Batérie sú umiestnené v držiaku BH383A, ktorý je zobrazený na obrázku 3.7. Z toho vyplýva, že kapacita zdroja závisí na použitých batériách.

Základné technické špecifikácie:

- držiak na batérie,
- 8x nabíjateľná batéria *Westinghouse* typu AA, 1,2 V DC,
- výsledné napätie 9,6 V DC,

- kapacita 2100 mAh.

Ako je opísané v kapitole 3.2.1, pre použité motory je odporúčané napájacie napätie 4,5 V DC. Preto bolo nutné maximálnu hodnotu tohoto napätia vhodným spôsobom obmedziť. Riešením tohto problému sa zaoberá kapitola 4.4.



*Obrázok 3.7. Držiak napájacích batérií.*



*Obrázok 3.8. Použitá batérie*

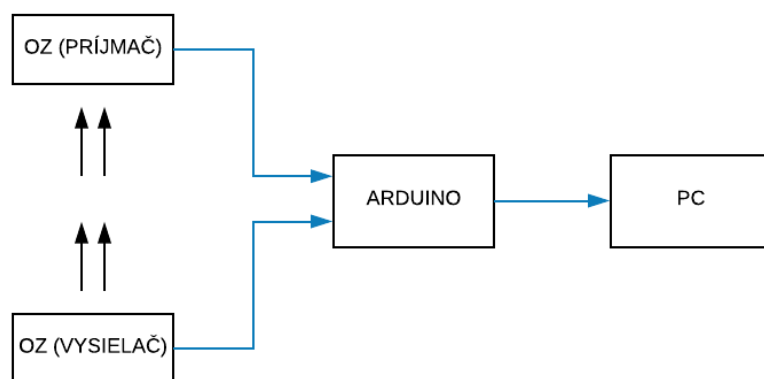
### 3.3 Časomiera

Pre meranie doby prejazdu autíčok jednotlivých tímov súťažnými dráhami bolo nutné vytvoriť systém, na ktorý sú kladené nasledujúce nároky:

- spoľahlivosť,
- vysoká presnosť,
- jednoduchá a rýchla obsluha,
- komponenty časomiere nemôžu mať vplyv na senzory a snímače použité v autíčkach,

- použitie Arduina pre spracovanie signálov zo snímačov ako ukážka flexibility tejto platformy,
- výsledné časy zobrazovať na monitore prenosného počítača.

Na základe týchto požiadaviek bol vykonaný prieskum trhu po ktorom bola ako snímač časomier zvolená optická závera FLAJZAR KF357, ktorá pracuje na princípe vysielачa a prijímača. Tento návrh zobrazuje bloková schéma na obrázku 3.9.



Obrázok 3.9. Boková schéma časomier

Vysielač obsahuje LASER, ktorý vysiela svetelný lúč dopadajúci na fototranzistor LTR4206 umiestnený na prijímacej časti. Pri dopade svetelného lúča na tento fototranzistor kontaktné relé umiestnené taktiež na strane prijímača je v zopnutom stave. Pri prerušení svetelného lúča, dôjde k rozopnutiu relé a tým prerušeniu signálu, ktorý je posielaný do Arduina. Pri vhodnom naprogramovaní dosky Arduino a správnej inštalácii senzorov na súťažné dráhy, získame jednoduchý systém splňujúci všetky vyššie uvedené požiadavky. Praktická realizácia časomier je opísaná v kapitole 4.5.

Základné technické parametre optickej závery:

- Napájacie napätie 12V,
- Max. odber prúdu: 65 mA,
- dosah závery cca 50 m.

## 4 Praktická realizácia

Podľa kapitoly 3 bola prevedená praktická realizácia autíčka. Na zostavený podvozok bolo nutné vhodne rozmiestniť ostatné komponenty tak, aby výsledný celok plnil účel, ktorým je demonštrovať realizovateľnosť zadania súťaže. Z toho vyplýva, že v praktickej časti bakalárskej práce nebolo dbané na dizajn autíčka, ale na jeho funkčnosť. Väčšina komponentov uľahčuje zostavenie autíčka napríklad aj tým, že podporujú jednotné konektory typu *header* s rozstupom 2,54 mm, čo znamená, že pri použití vhodných prepojovacích káblov s kompatibilným konektorom, sa pri realizácii projektu skracuje čas prepojenia komponentov medzi sebou.

### 4.1 Zostavenie autíčka

Nasledujúca časť práce sa zaoberá podrobným popisom vyhotovenia autíčka, návrhom programu pre komponenty, ktoré je pre ich správnu funkčnosť nutné naprogramovať a tvorbou mobilnej aplikácie pre MT s OS Android, ktorý slúži ako diaľkové ovládanie autíčka.

Na nasledujúcom obrázku 4.1 je zobrazená schéma zapojenia všetkých komponentov do výsledného celku. Voľbou a opisom týchto komponentov sa zaoberá kapitola 3.



Arduino a je prepojený so zabudovaným pull-up rezistorom, čo spôsobuje udržanie logickej úrovne 0 alebo 1 podľa aktuálnej polohy prepínača a tak Arduino dostáva informáciu o tom, či má vykonávať časť programu pre súťažnú *dráhu bludisko* alebo pre *sledovanie čiary*. Kód slúžiaci na prepínanie medzi týmito dvoma režimami zobrazuje výpis 1.

```
void loop()
{
  isSwitch = digitalRead(isSwitchPin);
  if (isSwitch)
  {
    Follower(); //Calling code for Follower
  }
  if (!isSwitch)
  {
    Labyrinth(); //Calling code for Labyrinth
  }
}
```

Výpis 1. Ukážka kódu slúžiaceho pre prepínanie režimu autička.

Zároveň so zmenou polohy prepínača sa privedie napájacie napätie buď do WiFi modulu, alebo do IR modulu vid' obrázok 4.1. Výhodou takéhoto riešenia je, že odpadá nutnosť prehrávania programu do Arduina pri zmene režimu pre jednotlivé súťažné dráhy. Spoločne s tým sa dosiahlo šetrenia energie vďaka odpájaniu nevyužívaných komponentov - v režime bludisko sa nevyužíva IR modul a v režime sledovania čiary zasa WiFi modul.

Ďalej je na schéme možné všimnúť si kondenzátor značený ako C1, pripojený paralelne k napájaciemu zdroju. Funkciou tohto kondenzátora je obmedziť podpäťtia vznikajúce vplyvom krátkodobých úbytkov napätia pri zmene zaťaženia akčných členov. Pre obmedzovanie spomínaného podpäťtia bolo vytvorené aj softvérové riešenie, ktoré zobrazuje výpis 2.

```
void RunForwardLabyrinth()
{
  motor1.run(FORWARD);
  motor2.run(FORWARD);
  motor3.run(FORWARD);
  motor4.run(FORWARD);
  int i;
  for (i = 0; i < forward_speedLabyrinth; i++)
  {
    motor1.setSpeed(i);
    motor2.setSpeed(i);
    motor3.setSpeed(i);
    motor4.setSpeed(i);
    delay(1);
  }
}
```

Výpis 2. Obmedzenie napätových špičiek pri rozbehu motorov

Pomocou cyklu „for“ súčiniteľ striedy PWM inkrementálne narastá od hodnoty 0 až po hodnotu zadanú v aplikácii na MT. Príkaz „delay(1)“ spôsobuje zmenu súčiniteľa striedy o hodnotu 1 za 1 ms.

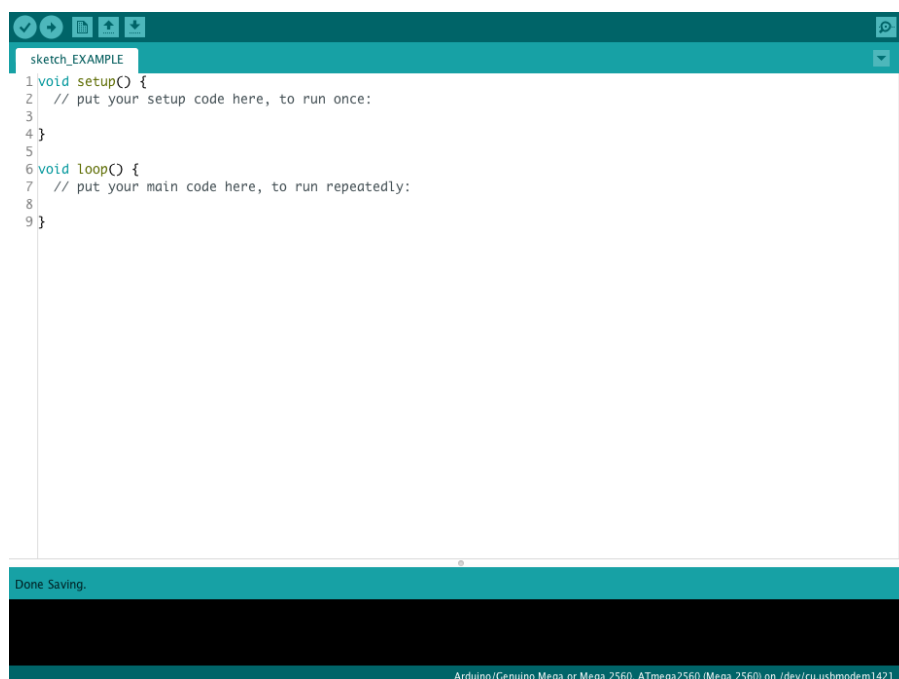
Po montáži a zapojení všetkých komponentov podľa schémy, bolo pre úspešné splnenie zadania nutné naprogramovať systémy, ktoré si to vyžadovali. Tomuto bodu sa venuje nasledujúca časť kapitoly.

## 4.2 Vývoj aplikačného softvéru

Ako programovací nástroj pre naprogramovanie komponentov ktoré si to vyžadujú, bol vo všetkých prípadoch použitý softvér *Arduino IDE*<sup>1</sup> v najnovšej verzii 1.8.5.

Jedná sa o *open-source* Arduino softvér (IDE), ktorý umožňuje písanie kódu a následne jeho nahrávanie do dosky. Je vytvorený pre počítačové platformy Windows, Mac OS X a Linux. Prostredie je napísané v jazyku Java a je založené na *Processing* a inom otvorenom softvéri. Tento softvér môže byť použitý s ktoroukoľvek doskou Arduino, ale aj inými doskami tretích strán. Existuje aj možnosť použiť online editor, ktorý je dostupný aj s cloud službou, kde je možné ukladať vytvorený kód, takzvaný sketch. [9]

Ukážka vývojového prostredia Arduino IDE je zobrazená na obrázku 4.2.



Obrázok 4.2. Vývojové prostredie Arduino IDE.

<sup>1</sup> <https://www.arduino.cc/en/Main/Software>

Ako môžeme vidieť na ukážke, toto prostredie je veľmi jednoducho členené. V hornej časti sa nachádzajú tlačidlá pre prácu s vytvoreným kódom. Konkrétne sú to (zľava): kompilovanie kódu, nahrávanie kódu do dosky, vytvorenie nového sketch-u, otvorenie existujúceho sketch-u a uloženie aktuálne editovaného sketch-u. V hlavnej časti je priestor pre editáciu kódu. Všimnime si, že pri otvorení okna pre vytváranie nového kódu, vývojové prostredie automaticky vypíše základné funkcie potrebné pre programovanie dosiek s mikrokontrolérom.

Prvá funkcia „void setup()“ slúži na prvotné nastavenia, ktoré má mikrokontrolér vykonať pre správnu funkciu výslednej aplikácie. Táto akcia sa vykoná pri každom zapnutí dosky. Jedná sa napríklad o príkazy pre spustenie komunikácie po sériovej linke, ktorá prebieha medzi programovanou doskou a počítačom prípadne inou doskou. Ďalší výborný príklad je nastavovanie I/O pinov, konkrétne či sa jedná o vstupné alebo výstupné piny. Každý pin môže plniť jednu z týchto funkcií, nie obe naraz.

Druhou funkciou je „void loop()“. V tejto funkcii sa cyklicky vykonáva hlavná časť kódu. To znamená, že sa v nej nachádzajú samotné príkazy pre akcie, ktoré má vykonať doska s mikrokontrolérom, ako napríklad meranie času na základe podnetu zo vstupnej periférie a podobne.

V pravej časti vývojového prostredia sa nachádza tlačidlo pre spustenie sériového monitora. Tento nástroj je veľmi užitočná pomôcka pri programovaní dosiek s mikrokontrolérom. Jeho vlastnosťou je schopnosť komunikovať s doskou pripojenou k počítaču pomocou sériovej linky USB. To znamená, že sa jedná o prostredie, prostredníctvom ktorého sú na monitore počítača zobrazované inštrukcie prijímané z pripojenej dosky a naopak, inštrukcie z počítača môžu byť zapísané do dosky.

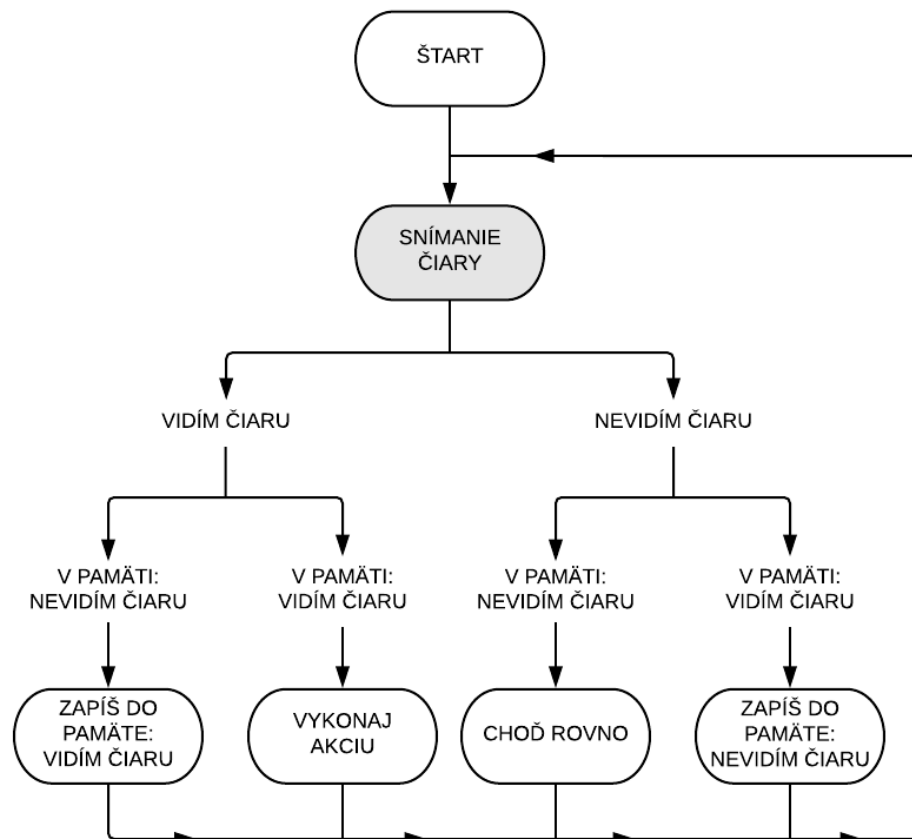
Nasledujúca časť kapitoly popisuje návrh algoritmov podľa zadania a samotné programovanie jednotlivých dosiek, použitých pre tvorbu výsledného autíčka.

#### **4.2.1 Návrh algoritmu pre sledovanie čiary**

Pri návrhu autíčka pre súťažnú dráhu „sledovanie čiary“ bol na základe experimentálneho zisťovania zvolený algoritmus, ktorý v každom cykle porovnáva aktuálny stav jednotlivých snímačov so stavom týchto snímačov, ktorý nadobudli v predchádzajúcom cykle programu a na základe výsledku tohto porovnania, je vykonaná daná akcia. Pri použití tohto algoritmu je správanie autíčka postačujúce, na demonštráciu realizovateľnosti zadania.

Pre túto súťažnú disciplínu boli použité 4 snímače IR žiarenia, umiestnené na prednej časti autíčka. Vývojový diagram zobrazený na obrázku 4.3 vysvetľuje tento algoritmus pre každý z nich.





Obrázok 4.3. Vývojový diagram inštrukcií pre sledovanie čiary

Následne bolo nutné tento navrhnutý algoritmus previesť do funkčného kódu pre dosku Arduino Mega 2560. Ako zobrazuje výpis 3, na začiatku bol každému snímaču pridelený konkrétny pin. Ako pamäť do ktorej sa zapisuje aktuálny stav snímačov, ktorý sa porovnáva so stavom nadobudnutým v nasledujúcom cykle programu, bola použitá premenná typu boolean pre každý snímač. To znamená, že na začiatku kódu museli byť definované 4 premenné (pre každý snímač jedna), do ktorých sa ukladá stav týchto snímačov. Definíciu týchto premenných zobrazuje rovnako výpis 3.

```

int sensor1pin = 37; // pin for IR sensor 1 - 4
int sensor2pin = 36;
int sensor3pin = 38;
int sensor4pin = 39;
bool sensor1 = true; // IR sensor 1 - 4 actual state
bool sensor2 = true;
bool sensor3 = true;
bool sensor4 = true;
  
```

Výpis 3. Pomocné premenné pre IR snímače

Ďalej bolo nutné piny definovať vo funkcii „void setup()“ ako vstupy, čo zobrazuje výpis 4.

```
pinMode(sensor1pin, INPUT);  
pinMode(sensor2pin, INPUT);  
pinMode(sensor3pin, INPUT);  
pinMode(sensor4pin, INPUT);
```

*Výpis 4. Definícia pinov IR snímačov ako vstupy Arduina*

Po tomto bode bolo možné naprogramovať samotný vopred opisovaný algoritmus. Vo výpise 5 je zobrazená ukážka kódu tohoto algoritmu pre snímač IR 1, ktorého pozíciu na autíčku zobrazuje bloková schéma na obrázku 3.1.

```
//SENSOR 1  
if (!sensor1 && sensor2 && sensor3 && sensor4 && sensor1state) //to the line S1  
{  
    sensor1state = false;  
}  
  
if (!sensor1 && sensor2 && sensor3 && sensor4 && !sensor1state) //on the line S1  
{  
    RotateRightFollower();  
}  
  
if (sensor1 && sensor2 && sensor3 && sensor4 && !sensor1state) //from the line S1  
{  
    sensor1state = true;  
}
```

*Výpis 5. Ukážka kódu algoritmu pre snímač IR1.*

Kód obsahuje tri krát funkciu „if()“. Prvá z týchto funkcií slúži na zapisovanie aktuálneho stavu snímača v prípade, kedy snímač v predchádzajúcom cykle programu nesnímal čiaru ale v aktuálnom cykle bol zmenený stav tohto snímača. Druhá slúži na vykonanie príslušnej akcie v prípade, kedy je snímačom snímaná čiaru a rovnaký stav nastal aj v predchádzajúcom cykle. Tretia funkcia je vykonávaná v prípade ak snímač v predchádzajúcom cykle programu snímal čiaru, ale v aktuálnom cykle túto čiaru nesníma. V tomto prípade bude prepísaná hodnota stavu snímača.

Každému snímaču pripadá definovaná akcia, ktorá ako vyplýva z výpisu 5, je volaná v druhej funkcii „if()“. Tieto akcie sú nasledovné:

- pre IR 1 je to rotácia vpravo,
- pre IR 2 autíčko spomalí motory na pravej strane a zrýchli motory na ľavej,
- pre IR 3 autíčko spomalí motory na ľavej strane a zrýchli motory na pravej,
- pre IR 4 je to rotácia vľavo.

Ukážka kódu volanej akcie pre snímač IR 1 je na nasledujúcom výpise 6.

```

void RotateRightFollower()
{
    motor1.run(FORWARD);
    motor2.run(FORWARD);
    motor3.run(BACKWARD);
    motor4.run(BACKWARD);
    motor1.setSpeed(150);
    motor2.setSpeed(150);
    motor3.setSpeed(150);
    motor4.setSpeed(150);
}

```

*Výpis 6. Volaná akcia pri snímaní čiary snímačom IR 1*

Ako vysvetľuje úvod kapitoly 3.2, z viacerých dôvodov je vhodné používať knižnice pre jednotlivé komponenty. Vo výpise 6 je možné vidieť použitie knižníc pre komponent *Adafruit Motor driver Shield L293D*, ktorý je výkonovou časťou autíčka, ktorá má za úlohu ovládať jednotlivé motory. Na základe dvoch jednoduchých príkazov pre každý motor, dokážeme udať smer otáčania motora a rýchlosť tohto otáčania. Viac o spôsobe riadenia jednotlivých motorov popisuje kapitola 4.4.

#### 4.2.2 Návrh algoritmu pre bludisko

Pri návrhu algoritmu pre zvládnutie súťažnej dráhy bludisko bolo nutné myslieť na fakt, že komunikácia prebieha medzi tromi komponentmi: MT, WiFi modulom a Arduino. Posledný zo spomenutých komponentov ich následne spracuje a zašle do výkonovej časti, tak ako to je zobrazené na blokovom diagrame na obrázku 3.1 a opísane v kapitole 3.1. Z toho vyplýva, že bolo nutné vytvoriť program zvlášť pre bezdrôtový člen (WiFi modul) a logický člen (dosku Arduino). Pre ovládací člen (MT s OS Android) bolo potrebné vytvoriť aplikáciu, pomocou ktorej bude autíčko ovládané, čiže aplikáciu, ktorá bude zasielať inštrukcie pre pohyb autíčka. Tvorba tejto aplikácie je opísaná v kapitole 4.3.

Pri WiFi module komunikáciu s MT vykonáva časť kódu zobrazená vo výpise 7.

```

void loop()
{
    server.handleClient(); //Listen for HTTP requests from Android App
    delay(10);
}

void HTTP_handleRoot(void)
{
    if (server.hasArg("State"))
    {
        Serial.println(server.arg("State")); //Send the variable State to Arduino via serial port
    }
    server.send(200, "text/html", "");
}

```

*Výpis 7. Príkazy pre sprostredkovanie komunikácie WiFi modulu s MT*

Tento kód zabezpečuje obsluhu pripojeného klienta (MT) prostredníctvom http protokolu, pričom WiFi modul je nastavený ako AP. V prípade prijatej žiadosti od klienta, ktorá obsahuje premennú *state*, zašle hodnotu tejto premennej po sériovej linke do pripojeného Arduina. Následne Arduino prijme tieto inštrukcie preposielané WiFi modulom a na základe toho, aká inštrukcia bola prijatá, vykoná príslušnú akciu. To akým spôsobom a aké inštrukcie zasiela MT do WiFi modulu opisuje kapitola 4.3.

Arduino prijíma a spracováva tieto inštrukcie pomocou kódu, ktorý je zobrazený na výpise 8.

```
void Labyrinth()
{
  if (Serial1.available() > 0)
  {
    command = Serial1.read();
    int icommand = command - 48; //ASCII conversion
    //Serial.println(icommand);
    if (icommand > 0 && icommand < 10)
    {
      forward_speedLabyrinth = icommand * speedCoefficient;
    }
    else
    {
      switch (command)
      {
        //default :RunStop();break;
        case 'F':
          RunForwardLabyrinth();
          break;
        case 'B':
          RunBackwardLabyrinth();
          break;
        case 'R':
          RotateRightLabyrinth();
          break;
        case 'L':
          RotateLeftLabyrinth();
          break;
        case 'S':
          RunStop();
          break;
      }
    }
  }
}
```

Výpis 8. Kód volaný v režime bludisko

### 4.3 Vývoj mobilnej aplikácie

Po prieskume trhu zameranom na vývoj mobilnej aplikácie pre OS Android, bol ako nástroj na tvorbu aplikácie zvolený *MIT App Inventor*.<sup>2</sup>

---

<sup>2</sup><http://appinventor.mit.edu/explore/>

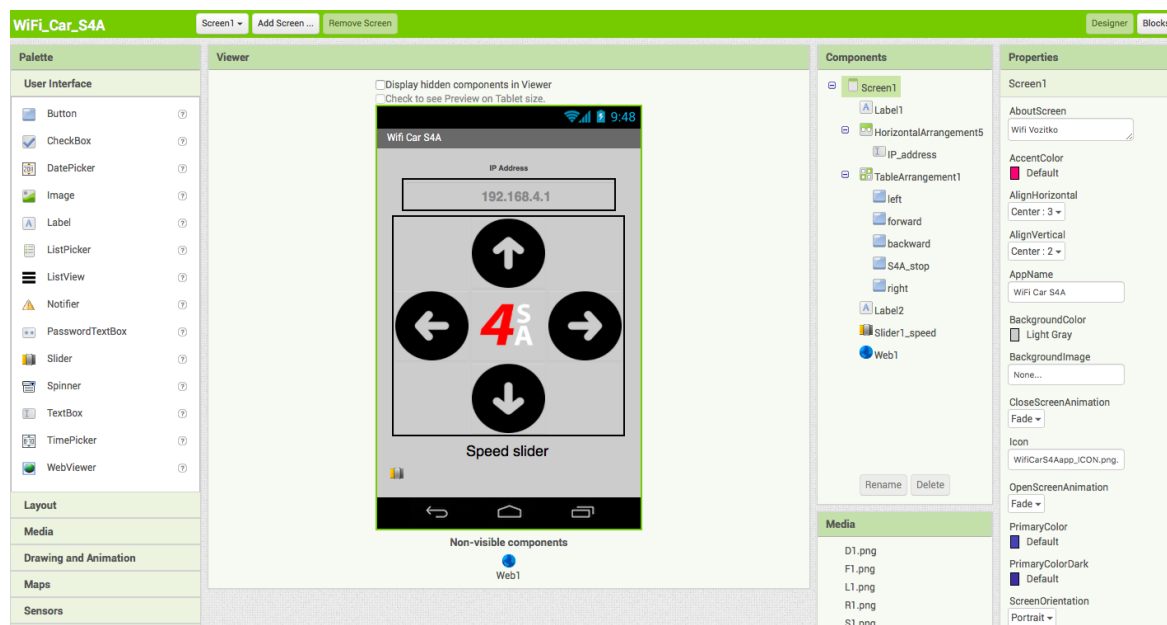
Jedná sa o *open-source* webovú aplikáciu pôvodne poskytovanou spoločnosťou *Google*, teraz spravovanou *Massachusetts Institute of Technology* (ďalej len MIT).

Umožňuje nováčikom počítačového programovania vytvárať softvérové aplikácie pre OS Android. Využíva grafické prostredie podobné Scratch-u a StarLogu TNG užívateľské prostredie, ktoré povoľuje užívateľom chytiť a umiestniť vizuálne objekty a tak vytvárať aplikácie, ktoré fungujú na zariadeniach s OS Android. [10]

Prostredie tohto nástroja je pomerne intuitívne. Delí sa na dve hlavné časti.

- a) *Designer* - v tomto okne sa navrhuje dizajn aplikácie. Objektom použitým v danej aplikácii sa pridelujú parametre, ako napríklad názov, pozícia, rozmery, obrázok a podobne.
- b) *Blocks* - v tomto okne je možné graficky programovať použité objekty.

Na obrázkoch 4.4 a 4.5 je možné vidieť vývojové prostredie oboch týchto okien.



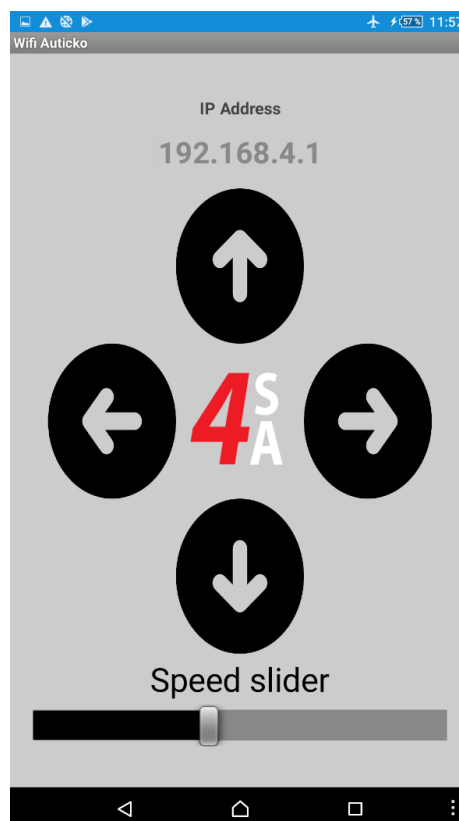
Obrázok 4.4. Vývojové prostredie MIT App Inventor - okno Designer



Obrázok 4.5. Vývojové prostredie MIT App Inventor - okno Blocks

Výhodou je fakt, že už vytvorené aplikácie sa dajú späťne importovať do *MIT App Inventor* a následne modifikovať. To umožňuje poskytnúť tímom overenú mobilnú aplikáciu, pomocou ktorej budú schopní nadviazať komunikáciu medzi aplikáciou a WiFi modulom *NodeMCU ESP8266 V3*.

Výslednú podobu užívateľského rozhrania mobilnej aplikácie pre diaľkové ovládanie autíčka je možné vidieť na obrázku 4.6. Vytvorená aplikácia s názvom *WiFi Car S4A* vo formáte *.apk* a *.aia* je priložená v prílohe C.



Obrázok 4.6. Výsledná podoba užívateľského rozhrania mobilnej aplikácie pre OS Android

Užívateľské rozhranie aplikácie je vytvorené tak, aby bolo čo najintuitívnejšie. V hornej časti obrazovky sa nachádza zobrazená aktuálna IP adresa, ktorá je manuálne nastavená pri tvorbe mobilnej aplikácie a musí byť zhodná s IP adresou WiFi modulu, ktorá je od výroby nastavená na 192.168.4.1.

Pre nadviazanie komunikácie mobilnej aplikácie s WiFi modulom autíčka, je nutné mať zapnutú WiFi na mobilnom telefóne a zároveň spustený WiFi modul, naprogramovaný podľa kapitoly 4.2.2. Ďalej stačí len vybrať vysielanú WiFi sieť autíčka zo zoznamu nájdených WiFi sietí v nastaveniach MT. Pre samotný pohyb autíčka slúžia 4 tlačidlá - šípky hore a dole určujú smer pohybu dopredu a dozadu. Šípky po stranách určujú rotáciu autíčka vpravo alebo vľavo. Každému z týchto prvkov bola priradená akcia, ktorá má byť vykonaná pri stlačení a následnom pustení daného prvku.

Ako zobrazuje obrázok 4.5, pri stlačení tlačidla, je na IP adresu prostredníctvom protokolu http zaslaná informácia, obsahujúca premennú *state*, ktorá pri stlačení jednotlivých tlačidiel nadobúda nasledujúce hodnoty:

- „F“ pri stlačení šípky hore,
- „B“ pri stlačení šípky dole,
- „R“ pri stlačení šípky vpravo,
- „L“ pri stlačení šípky vľavo.

Pri následnom pustení všetkých týchto tlačidiel, premenná *state* nadobúda hodnotu „S“. Tieto znaky následne prijme WiFi modul a zašle ich ďalej do Arduina tak, ako to je opísané v kapitole 4.2.2.

V spodnej časti obrazovky je umiestnený posuvný jazdec pre nastavenie aktuálnej rýchlosti autíčka. Ako vysvetľuje kapitola 4.4, maximálna hodnota striedy PWM v príkazoch pre dosku Arduino musela byť obmedzená na hodnotu 150. To znamená, že posunutie jazdca do pravej krajnej polohy musí zodpovedať nastaveniu striedy na túto hodnotu. Na základe tejto skutočnosti bolo nutné zvoliť, aké hodnoty budú zasielané pri posúvaní jazdca. Pri tom všetkom bolo nutné myslieť aj na to, že aplikácia posiela do WiFi modulu informácie v type čísla, no Arduino ich z modulu cez sériovú linku číta ako sled jednoznakových informácií. Preto je nutné každý prijatý znak pretypovať späť na numerickú hodnotu. Na to slúži časť kódu zobrazeného vo výpise 9.

```

{
  if (Serial1.available() > 0)
  {
    command = Serial1.read();
    int icommand = command - 48; //ASCII conversion

    if (icommand > 0 && icommand < 10)
    {
      forward_speedLabyrinth = icommand * speedCoefficient;
    }
  }
}

```

*Výpis 9. Kód pre načítanie informácie zo sériovej linky a následné pretypovanie podľa ASCII*

Tento výpis zobrazuje príkaz pre načítanie informácie zo sériovej linky. Pokiaľ je dostupná informácia väčšia ako 0 bude zapísaná do premennej *command*, ktorá je typu char. Keďže sa jedná o znakovú informáciu, tak napríklad pri zaslaní čísla 0 z mobilnej aplikácie, premenná *command* nadobudne znakovú hodnotu 0 čo je po prevode podľa ASCII<sup>3</sup> v skutočnosti numerická hodnota 48. Preto bolo nutné vytvoriť novú premennú, *icommand* typu integer, do ktorej je zapísaná hodnota premennej *command* znížená o číslo 48. Tak sa dospelo k získaniu skutočnej hodnoty, ktorú zasielala mobilná aplikácia.

Ako najjednoduchší spôsob sa teda javilo posielanie jednoznakových informácií. Preto pripadla voľba na celé čísla 2 až 5, čo zodpovedá minimálnej až maximálnej hodnote jazdca slúžiaceho pre zmenu rýchlosti. Zároveň sa tým dosiahlo aj to, že pri násobení týchto čísel číslom 30, je výsledok vždy celé číslo od 60 do 150. Tým bola splnená vyššie uvedená podmienka obmedzenia maximálnej hodnoty napájacieho napätia pre použité motory pri danom zdroji.

Je nutné poznamenať, že takto vytvorená aplikácia nepodporuje stlačenie dvoch a viac tlačidiel naraz.

#### **4.4 Finalizácia demonštračného autíčka**

Použitý zdroj jednosmerného napätia pri plnom nabití dodáva približne 10,6 V pričom odporúčané napájacie napätie pre použité motory je 4,5 V. V závere praktickej realizácie bolo nutné vyriešiť tento problém.

Výstupom výkonovej časti je riadiace napätie pre jednotlivé motory. Ako zobrazuje obrázok 4.7, toto napätie je pulzne-šírkovo modulované. Z toho vyplýva, že ako vhodná voľba obmedzenia maximálnej hodnoty riadiaceho napätia motorov, bolo obmedziť maximálnu hodnotu striedy tejto PWM. Táto PWM je vytváraná príkazmi pre dosku

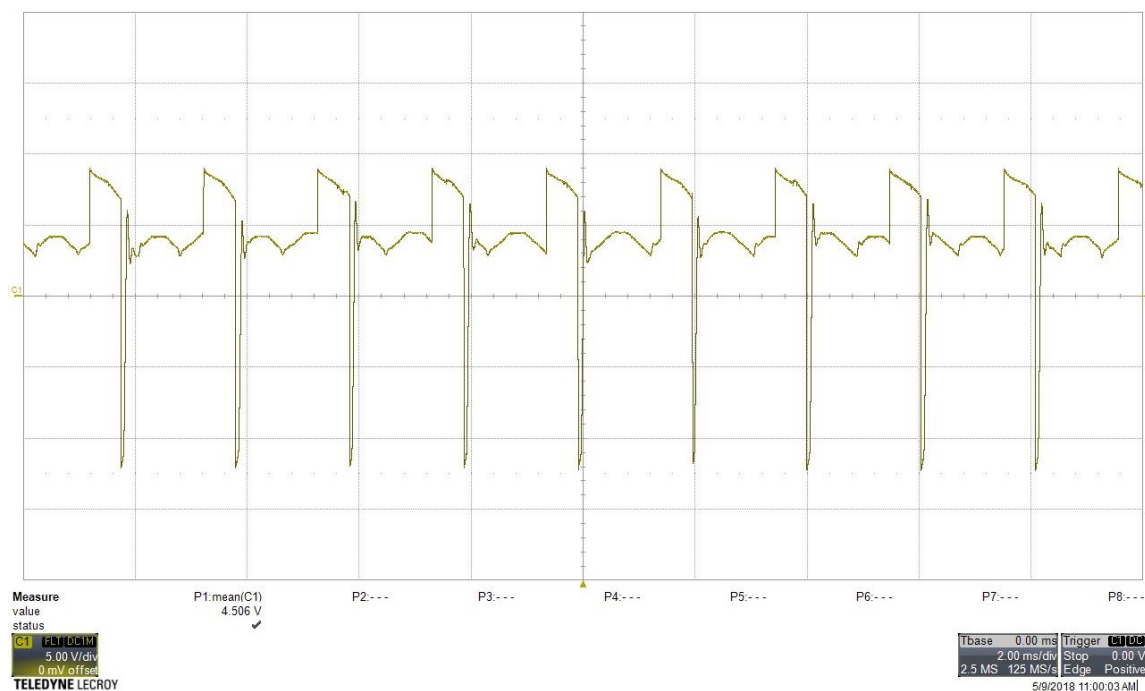
---

<sup>3</sup> <http://www.theasciicode.com.ar/ascii-printable-characters/capital-letter-f-uppercase-ascii-code-70.html>



Arduino Mega 2560, ktorá má 8-bitový PWM výstup, čo znamená, že tieto príkazy môžu nadobúdať hodnoty 0 až 255 čo zodpovedá striede PWM 0 až 100%.

Tu bolo nutné zvoliť maximálnu povolenú hodnotu striedy v kombinácii s použitým napájacím zdrojom. Nasledujúci graf na obrázku 4.6 bol nameraný pomocou osciloskopu a zobrazuje priebeh riadiaceho napätia motora M1. Ostatné motory sú napájané rovnakým napätím.



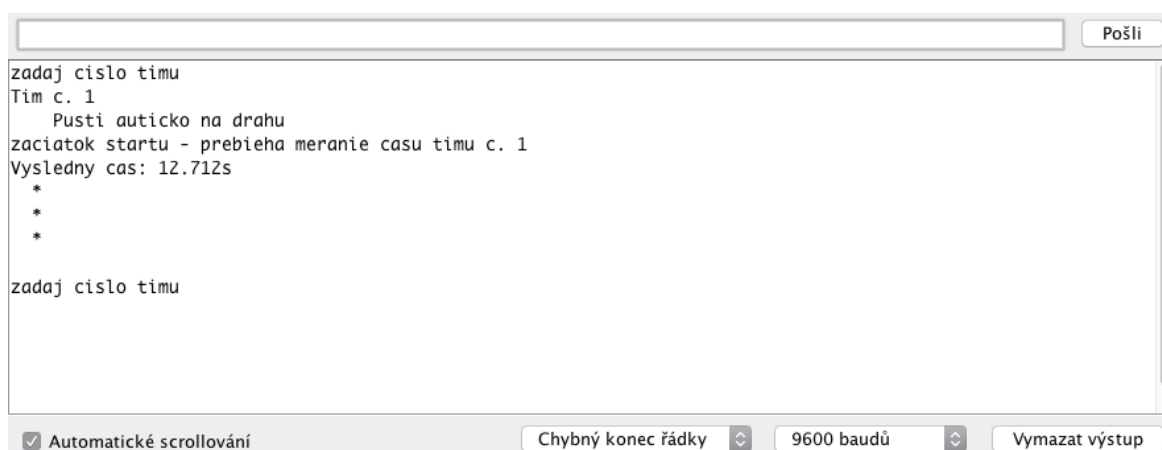
Obrázok 4.7. Priebeh riadiaceho napätia motora M1

V ľavej spodnej časti obrázka je možné všimnúť si strednú hodnotu napätia 4,506 V. Pri tomto meraní bola pomocou príkazu pre Arduino nastavená strieda PWM na hodnotu 27,5%, avšak je nutné dodať, že toto meranie bolo uskutočnené v prípade, kedy motor je zaťažovaný len hybnými silami prevodovky a kolesa. Na základe tohto merania a následného experimentálneho overovania bola stanovená maximálna strieda PWM na hodnotu 58,82%, čo pre príkaz pre Arduino zodpovedá hodnote 150.

## 4.5 Zhotovenie časomieri

Podľa návrhu opísaného v kapitole 3.3 bol vyhotovený systém určený pre meranie doby prejazdu súťažných autíčok jednotlivými dráhami. Optická závera bola dodaná vo forme stavebnice a tak bolo nutné najskôr ju poskladať. Takto zostavené komponenty spoločne s fotografiami zrealizovaného systému časomieri sú zobrazené v prílohe B. Pre výpočet času bolo nutné naprogramovať dva systémy Arduino - pre každú súťažnú dráhu jednu. Zdrojové kódy pre tieto systémy sa nachádzajú v prílohe C. Obrázok 4.8 zobrazuje ukážku

výpisu z prostredia sériového monitora pri meraní doby prejazdu autíčka súťažnou dráhou *Sledovanie čiary*.



Obrázok 4.8. Výpis časomier z prostredia sériového monitora

## **5 Vyhliadky do budúcnosti**

Zostavenie autíčka bolo prevedené tak, aby sa v nasledujúcich ročníkoch súťaže dalo pri tvorbe zadania nadviazať na platformu vytvorenú v rámci tejto bakalárskej práce. Myšlienkou je teda pokračovať v hlavných témach súťaže spomenutých v úvode práce a do koncepcie zapojiť nové komponenty ako sú napríklad senzor farieb a Bluetooth modul. Pomocou týchto súčiastok bude možné získavať informácie o napríklad farbách značiek rozmiestnených na súťažnej dráhe a následne tieto informácie zasielať na vzdialený server. Samozrejme je možné použiť aj iné dostupné komponenty.

## Záver

Cieľom bakalárskej práce bolo navrhnuť a zrealizovať demonštračné autíčko s riadiacim systémom Arduino a koncept súťaže pre nasledujúci tretí ročník súťaže Students for Automotive. S tým súvisel dôkladný prieskum trhu zameraný na možnosti platformy Arduino. Na základe toho bol vytvorený koncept zadania, pričom pri jeho tvorbe sa musel brať do úvahy fakt, že súťaž je určená žiakom stredných škôl a teda zložitosť vytvoreného zadania musí byť úmerná ich schopnostiam. Zároveň sa chcela zachovať myšlienka motivovať tímy k tvorbe vlastného viacúčelového projektu a tým im ukázať aké rozmanité možnosti ponúka spomínaná platforma. Práve preto bolo rozhodnuté vytvoriť zadanie, ktoré bude žiakov baviť, no zároveň ich učiť pre nich nové poznatky z oblasti elektrotechniky a informačných technológií. Jedná sa o autíčko, ktoré má byť schopné vykonať dve úlohy a to sledovanie čiary a prechod bludiskom pomocou bezdrôtového ovládania mobilným telefónom. Takéto zadanie má síce vyššie nároky na schopnosti zapojených tímov, a preto ďalšími výstupmi tejto bakalárskej práce sú mobilná aplikácia, ktorú tímy dostanú k dispozícii a manuál, ktorý im má pomôcť s hlavnými problémami pri skladaní ich projektu.

Vytvorený manuál ich prevedie krok po kroku tak, aby na jeho konci úspešne nadviazali komunikáciu medzi mobilným telefónom a autíčkom a tak vyriešili najväčšie problémy pri riešení zadania súťaže. Zároveň obsahom manuálu je kapitola vysvetľujúca ako jednoducho sa dá modifikovať dostupná aplikácia podľa predstáv jednotlivých tímov.

Vytvorené demonštračné autíčko, ktoré je výstupom tejto práce je plne funkčné a je schopné splniť zadanie súťaže, pričom nebolo dbané na to, aby toto zadanie splnilo čo najlepšie. Tým sa dosiahlo faktu, že tímom sa naskytuje možnosť prekonať schopnosti demonštračného autíčka. Praktickým zhotovením autíčka sa naplnil hlavný cieľ práce a to, že autíčko je ukážkou splniteľnosti vytvoreného zadania súťaže.

## Zoznam použitej literatúry

- [1] *Introduction* [online]. In: . [cit. 2018-05-21]. Dostupné z:  
<https://www.arduino.cc/en/Guide/Introduction>
- [2] O Arduinu. VODA, Zbyšek a TÝM HW KITCHEN. *Průvodce světem Arduina*. Druhé vydanie. Bučovice: Martin Stříž, 2017, s. 1. ISBN 978-80-87106-93-8.
- [3] *Multi-Chassis - 4WD Kit (Basic)* [online]. [cit. 2018-05-21]. Dostupné z:  
<https://www.sparkfun.com/products/12089>
- [4] *Arduino Mega 2560 Rev3* [online]. In: . [cit. 2018-05-21]. Dostupné z:  
<https://store.arduino.cc/arduino-mega-2560-rev3>
- [5] *Motor shield L293D* [online]. In: . [cit. 2018-05-21]. Dostupné z:  
<http://navody.arduino-shop.cz/navody-k-produktum/motor-shield-l293d.html>
- [6] *NodeMCU a jeho verzie: doska s Wi-Fi čipom ESP8266* [online]. In: . [cit. 2018-05-21]. Dostupné z: <https://www.root.cz/clanky/nodemcu-a-jeho-verzie-doska-s-wi-fi-cipom-esp8266/>
- [7] *Senzor překážek YL-70* [online]. In: . [cit. 2018-05-21]. Dostupné z: <http://navody.arduino-shop.cz/navody-k-produktum/senzor-prekazek-pro-arduino-yl-70.html>
- [8] *Módulo Seguidor de Línea Ajustable de 4 Canales Modelos YL-70 e YL-73* [online]. [cit. 2018-05-21]. Dostupné z: <http://www.maxelectronica.cl/luz-color/374-modulo-seguidor-de-linea-ajustable-de-4-canales-modelos-yl-70-e-yl-73.html>
- [9] *Arduino IDE 1.8.5* [online]. In: . [cit. 2018-05-21]. Dostupné z:  
<https://www.arduino.cc/en/Main/Software>
- [10] *The MIT roots of Google's new software* [online]. In: . [cit. 2018-05-21]. Dostupné z:  
<http://news.mit.edu/2010/android-abelson-0819>

## Zoznam použitých obrázkov

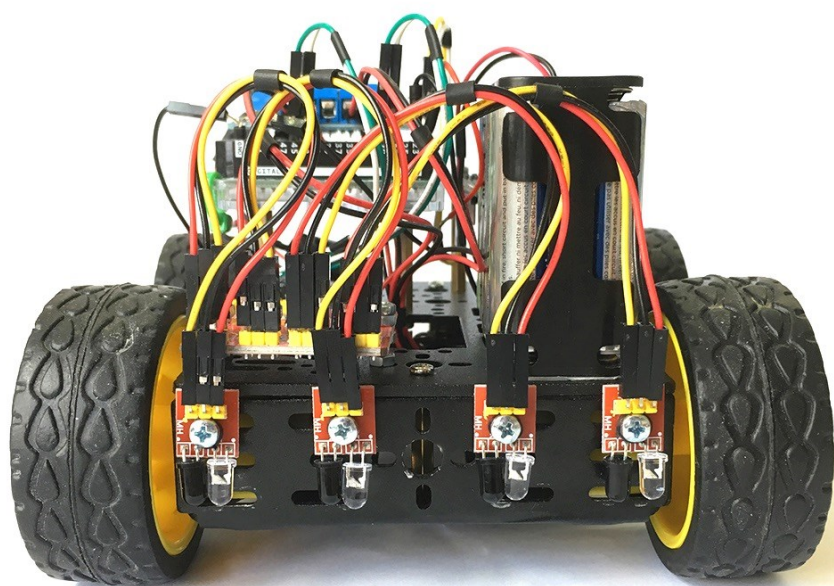
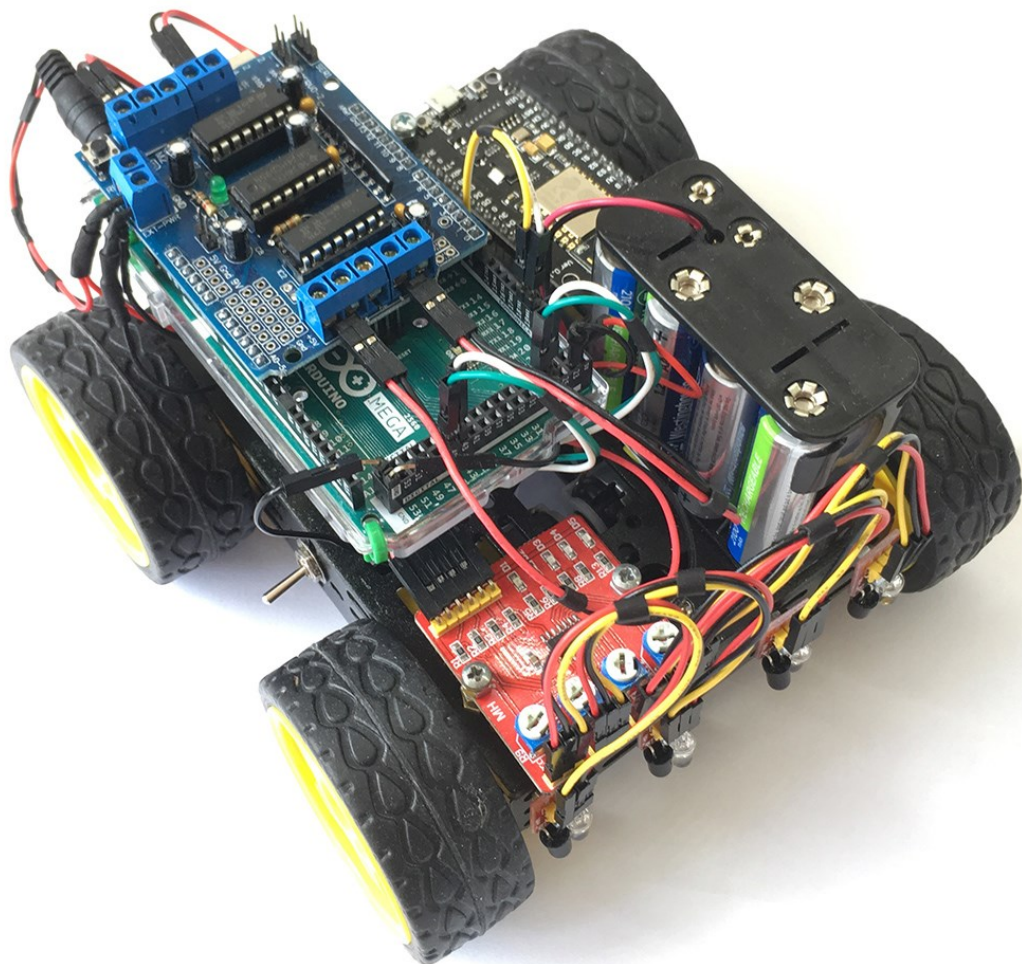
Obrázok 1.1. Online Dotazník - otázka č.1.....	11
Obrázok 1.2. Online Dotazník - otázka č.2.....	12
Obrázok 1.3. Online Dotazník - otázka č.3.....	12
Obrázok 1.4. Online Dotazník - otázka č.4.....	13
Obrázok 1.5. Online Dotazník - otázka č.5.....	13
Obrázok 2.1. Súťažná dráha - sledovanie čiary (Masarykov okruh).....	15
Obrázok 2.2. Súťažná dráha - bludisko .....	16
Obrázok 3.1. Bloková schéma navrhnutého demonstračného autíčka .....	18
Obrázok 3.2. Použitý podvozok 4WD .....	20
Obrázok 3.3. Arduino Mega 2560 .....	22
Obrázok 3.4. Adafruit Motor Driver Shield .....	23
Obrázok 3.5. NodeMCU ESP8266.....	24
Obrázok 3.6. Senzor prekážok YL-70 .....	25
Obrázok 3.7. Držiak napájacích batérií. ....	26
Obrázok 3.8. Použité batérie.....	26
Obrázok 3.9. Boková schéma časomier .....	27
Obrázok 4.1. Schéma zapojenia komponentov autíčka .....	29
Obrázok 4.2. Vývojové prostredie Arduino IDE.....	31
Obrázok 4.3. Vývojový diagram inštrukcií pre sledovanie čiary .....	33
Obrázok 4.4. Vývojové prostredie MIT App Inventor - okno Designer .....	37
Obrázok 4.5. Vývojové prostredie MIT App Inventor - okno Blocks.....	38
Obrázok 4.6. Výsledná podoba užívateľského rozhrania mobilnej aplikácie pre OS Android.....	38
Obrázok 4.7. Priebeh riadiaceho napätia motora M1 .....	41
Obrázok 4.8. Výpis časomier z prostredia sériového monitora.....	42

## **Zoznam výpisov zdrojového kódu**

Výpis 1. Ukážka kódu slúžiaceho pre prepínanie režimu autíčka. ....	30
Výpis 2. Obmedzenie napät'ových špičiek pri rozbehu motorov .....	30
Výpis 3. Pomocné premenné pre IR snímače .....	33
Výpis 4. Definícia pinov IR snímačov ako vstupy Arduina .....	34
Výpis 5. Ukážka kódu algoritmu pre snímač IR1 .....	34
Výpis 6. Volaná akcia pri snímaní čiary snímačom IR 1 .....	35
Výpis 7. Príkazy pre sprostredkovanie komunikácie WiFi modulu s MT.....	35
Výpis 8. Kód volaný v režime bludisko .....	36
Výpis 9. Kód pre načítanie informácie zo sériovej linky a následné pretypovanie podľa ASCII .....	40

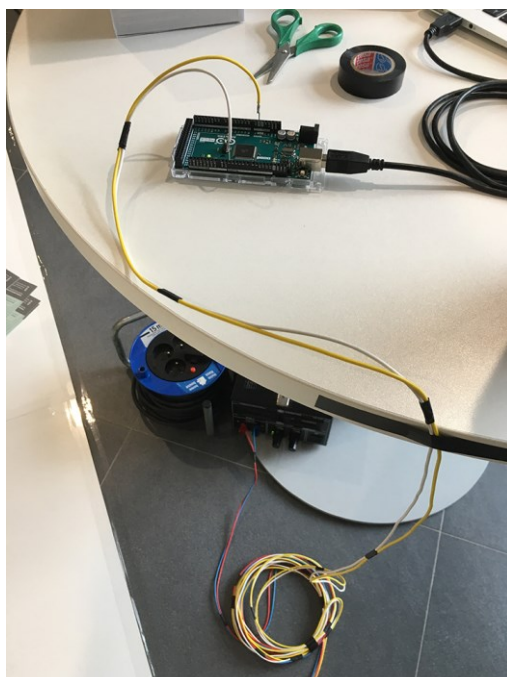
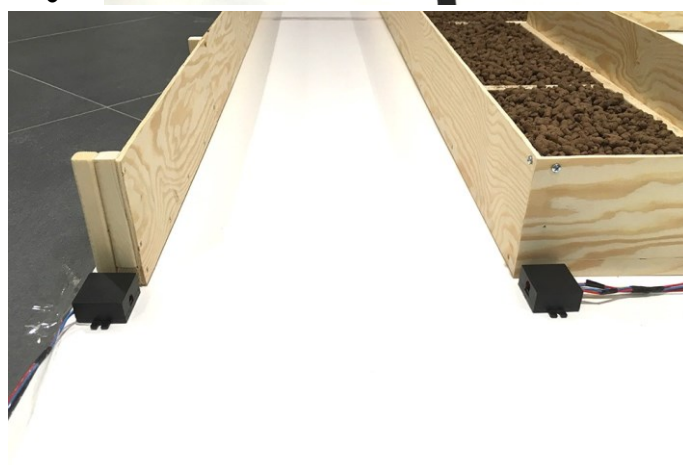
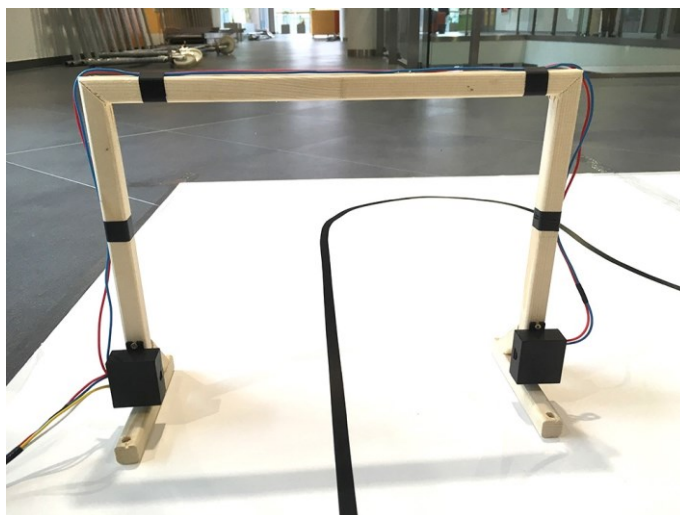
## Prílohy

### A Fotografie autíčka





## B Fotografie časomíry



## **C Obsah CD**

K práci priložené CD obsahuje nasledujúce súbory.

1. Text BP
  - BP\_CER0351.pdf
2. Manuál pre tímy v českom jazyku
  - Manual\_S4A.pdf
3. Inštalačný súbor aplikácie pre OS Android
  - WiFi Car S4A.apk
4. Projektový súbor aplikácie pre vývojové prostredie MIT App Inventor
  - WiFi Car S4A.aia
5. Zdrojové kódy pre použité komponenty
  - S4A\_NodeMCUESP\_8266.ino
  - S4A\_ArduinoMega2560.ino
  - S4A\_CasomieraBludisko.ino
  - S4A\_Casomiera\_SledovanieCiary.ino